JC914 U.S. PTO
09/663701
09/15/00

Patent Office
Canberra

I, KAY WARD, TEAM LEADER EXAMINATION SUPPORT AND SALES hereby certify that annexed is a true copy of the Provisional specification in connection with Application No. PQ 2912 for a patent by SILVERBROOK RESEARCH PTY LTD filed on 17 September 1999.

WITNESS my hand this
Sixth day of April 2000

KWard

KAY WARD
TEAM LEADER EXAMINATION
SUPPORT AND SALES

AUSTRALIA

PATENTS ACT 1990

PROVISIONAL SPECIFICATION

FOR THE INVENTION ENTITLED:-

"A SELF MAPPING SURFACE AND RELATED APPLICATIONS"

The invention is described in the following statement:-

The present inventions relate to a self mapping surface and other related applications as described herein.

The inventions have been developed primarily for use with a printer which interacts with the Internet or World Wide Web and will be described hereinafter predominantly with reference to that application. However, the inventions are not limited to that particular field of use.

While the following detailed description is of specific embodiments of the inventions, it will be appreciated by those skilled in the art that these inventions may be embodied in many other respective forms. Such forms include the numerous novel features, novel interactions and novel combinations of features and interactions that are described in the accompanying Appendices.

The specific embodiments of the inventions are disclosed in the accompanying documents that are respectively identified as:

Appendix 1, Netpage, System Overview, draft version 0.3;

Appendix 2, Netpage, Applications, draft version 0.1;

Appendix 3, Netpage Printer, Design Description, draft version 0.3; and

Appendix 4, S-print Product Concept, draft version, 0.2.

DATED this 17th day of September, 1999
SILVERBROOK RESEARACH PTY LTD

Attorney: JOHN B. REDFERN
Fellow Institute of Patent Attorneys of Australia
of BALDWIN SHELSTON WATERS

Appendix 1
(Pages 60)

Netpage
System Overview

draft version 0.3, 7 September 1999

Silverbrook Research Pty Ltd
393 Darling Street, Balmain
NSW 2041 Australia
Phone: +61 2 9818 6633
Fax: +61 2 9818 6711
Email: info@silverbrook.com.au

Confidential

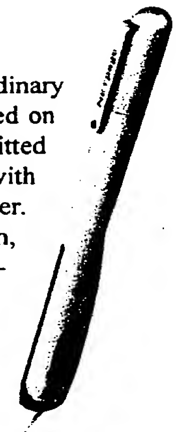
Document History

Version	[illegible]	[illegible]	[illegible]
0.3d	7 September 1999	Paul Lapstun Kia Silverbrook	Split Netpage System Design Description into multiple documents. Refined object models. Added application detail.
0.2d	1 June 1999	Paul Lapstun Kia Silverbrook	Initial draft issue.

Contents

OVERVIEW

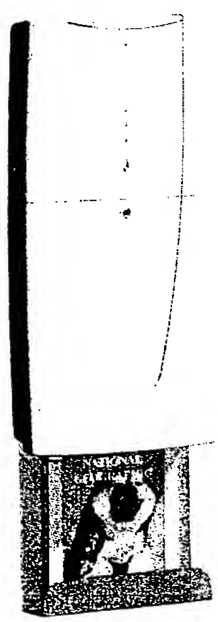
1 Introduction



Netpages are pages of high-quality text, graphics and images printed on ordinary paper, but which work almost like interactive Web pages. Information encoded on each page in invisible ink is picked up by an optically-imaging pen and transmitted to the network. Active "links" and "buttons" on each page can be "pressed" with the pen to request information from the network or signal preferences to a server. Text written by hand on a Netpage is automatically recognized via the pen, allowing forms to be filled in. Signatures recorded on a Netpage are automatically verified, allowing e-commerce transactions to be securely authorized.

The pen, shown on the right, works in conjunction with a Netpage Printer, an Internet-connected printing appliance for home, office or mobile use. The pen is wireless and communicates with the Netpage Printer using an encrypted radio frequency signal.

The Netpage Printer delivers, periodically or on demand, personalized newspapers, magazines, catalogs, brochures and other publications, all printed at high quality as interactive Netpages. Unlike a personal computer, the Netpage Printer, shown on the right, is an appliance typically wall-mounted in the kitchen or near the breakfast table, i.e. the place where the morning news is first consumed, and the household's point of departure for the day. It also comes in tabletop, desktop, portable and miniature versions.



Netpages printed at their point of consumption combine the ease-of-use of paper with the timeliness and interactivity of an interactive medium.

Netpages are crucially enabled by Memjet printing technology [1], which makes high-speed magazine-quality printing affordable to consumers. A Netpage publication has the physical characteristics of a traditional newsmagazine, i.e. a set of letter-size glossy pages printed in full color on both sides, bound together for easy navigation and comfortable handling. The Netpage Printer prints 60 to 90 full-color Netpages per minute.

The Netpage Printer exploits the growing availability of broadband Internet access. Cable service is available to 95% of households in the United States [56], and cable modem service offering broadband Internet access is already available to 20% of these [8]. The Netpage Printer also operates with slower connections, but with longer delivery times and lower image quality.

Netpage Publication Servers on the Internet deliver print-quality publications to Netpage Printers. Periodical publications are delivered automatically to subscribing Netpage Printers via pointcasting and multicasting Internet protocols. Personalized publications are filtered and formatted according to individual user profiles.

A Netpage Printer supports any number of pens, and a pen can work with any number of Netpage Printers. Each Netpage Pen has a unique identifier. A household may have a collection of colored Netpage Pens, one assigned to each member of the family. This allows each user to maintain a distinct profile with respect to a Netpage Publication Server. A Netpage Pen can also be registered with a Netpage Registration Server and linked to one or more payment card accounts. This allows e-commerce payments to be securely autho-

rized using the Netpage Pen. The Netpage Registration Server compares the signature captured by the Netpage Pen with a previously registered signature, allowing it to authenticate the user's identity to the e-commerce system. Other biometrics can also be used to verify identity. A version of the Netpage Pen includes fingerprint scanning, verified in a similar way by the Netpage Registration Server.

Although a Netpage Printer delivers periodicals such as the morning newspaper without user intervention, it never delivers unsolicited junk mail. It only delivers periodicals from subscribed or otherwise authorized sources. The Netpage Printer is unlike a fax machine or e-mail account which is visible to any junk mailer who knows the telephone number or email address.

1.1 PUBLISHING TO NETPAGE PRINTERS

There are a number of advantages to publishing to Netpage Printers. The magazine quality of Netpage Printer output makes it a more attractive publishing and advertising medium than both traditional newsprint and computer screens.

The cost of paper and ink consumption is transferred to the user. Subscription fees can be eliminated entirely in lieu of the user taking on this extra cost, and the user thus perceives a better-value product. The erratic price of newsprint is removed from publishers' profit equations, resulting in more stable margins. A new market for paper and ink consumables, with its own margins, is created.

The cost of consumables can be selectively subsidized, for example when non-editorial publications such as product brochures and account statements are printed.

Capital and maintenance expenditure on printing plant is effectively transferred to the user, although the perceived expense is small because Netpage Printers are sold at close to cost or given away to encourage adoption, subsidized by future advertising profits. Maintenance can also be subsidized or its cost included in a longer term service agreement.

Costly physical distribution is replaced by electronic distribution via a preexisting and widely subscribed network - the Internet.

Both the editorial and advertising content of publications delivered via the Netpage Network can be customized for each user. Editorial content can be personalized according to the user's profile. Advertising can be localized to the user's locality and optionally targeted to the user's demographic.

A personalized publication can be a small fraction of the size of its traditionally-delivered counterpart, yet contain the same amount of information relevant to the user, and in a more accessible form. The user appreciates the more efficient and digestible publication.

Localized advertising can be targeted to more specific localities and their associated demographics, and this allows advertising space to be exploited more efficiently, i.e. with less waste. Advertisers are constantly pressing traditional publishers for greater localization, something which they have great difficulty delivering cost-effectively.

By revealing personal information such as age, gender, marital status, income, profession, education, etc., the user can allow the advertising to be more carefully targeted. In return they can receive greater subsidies and discounted product offers. As advertising becomes more targeted, it becomes less of a nuisance and more of a service in itself.

Although the publication's per-page circulation figures fall drastically, the publication's actual per-section readership is preserved, and the correspondingly higher advertising rates for personalized delivery can exactly compensate for this.

Advertising delivered via the Netpage Network has the dual benefits of print and online delivery. Print supports the impact of large-format ads. Online delivery supports customization, linking, and measurability, and consequently online charging models.

Consider a full-page advertisement for a new car model in a newsmagazine delivered via the Netpage Network. The advertising campaign can be national or even international. The ad only appears if compatible with the user's demographic, either implied by their ZIP Code or more explicitly by their personal details. Anyone who requests a product brochure via the on-ad button receives one immediately via their Netpage Printer, customized with a list of local dealers. If they press a particular dealer's "contact me" button in the brochure, the dealer receives a message via the system and contacts the user by telephone.

The publisher profits in the normal way by selling the advertising space, but can also profit by receiving a fee on the "click-through" to the brochure, and a commission on any product sale which eventuates.

The Netpage Network promises to be the most effective advertising medium ever conceived. It combines the editorial and print quality of traditional publications with arbitrarily finely targeted advertising, and provides a direct link between advertising, product information, and purchasing. Added revenue from click-through fees and e-commerce commissions may even allow users' costs - printer, ink, paper, and Internet access - to be fully subsidized.

2 The Demise of Paper

Online publication has many advantages over traditional paper-based publication. From the consumer's point of view, information is available on demand, information can be navigated via hypertext links, information can be searched, and information can be automatically personalized.

From the publisher's point of view, the costs of printing and physical distribution are eliminated, and the publication becomes more attractive to the advertisers who pay for it because it can be targeted to specific demographics and linked to product sites.

Online publication also has a few disadvantages. Computer screens are inferior to paper. At the same quality as a magazine page, an SVGA computer screen displays only about a fifth as much information[1]. Both CRTs and LCDs have brightness and contrast problems, particularly when ambient light is strong. Ink on paper, being reflective rather than emissive, is both bright and sharp in ambient light.

Faced with reading more than the most trivial amounts of text on a screen, most people prefer to print it before reading it. Increasingly, online publishers are recognizing this and providing information in formats suitable for printing. At one extreme this means providing text-only versions of documents so they print efficiently, i.e. without imposing a screen format on the printed page; at the other extreme it means providing formatted versions of documents - e.g. in Adobe's Portable Document Format (PDF) - so they print at high quality.

Editorial content is often compromised to fit the online medium and the habits of the people who frequent it, who tend to browse rather than read. Although powerful new advertising models become possible, it becomes more difficult to deliver effective advertisements.

To truly enable online publication, many people envisage a universal information appliance - a lightweight portable "tablet" with a page-size touch-sensitive color display and a high-bandwidth wireless connection to the Internet. First proposed by Xerox PARC's Alan Kay in the mid 1970s in the form of the "Dynabook", and partially realized in recent paperback-sized electronic books, even Bill Gates is now confidently predicting that such a device will soon augur the death of paper publications [24].

To achieve low power consumption, low weight, and paper-like display quality, a bistable reflective display technology is required. Several candidates are now emerging from the labs, including Kent Display's cholesteric LCD technology (chLCD) [35], Xerox' "Gyricon" rotating ball technology [27], and E Ink's electrophoretic technology [20,46]. ChLCD is arguably closest to practical deployment [19].

Next-generation cellular phone networks promise 2Mbps packet switching [21], comparable to the broadband access people are getting used to in cable networks [43]. Satellite networks, while offering or promising still higher speeds [28,49,62], require receivers difficult to deploy in mobile devices.

Beyond the vision of the basic tablet, E Ink imagines its digital ink "printed" onto a number of flexible pages bound into a book, preserving the physical navigability of a paper-based publication, and approaching its low cost, but allowing the pages to be rewrit-

1. Assuming a magazine page has an equivalent digital resolution of 200 continuous-tone pixels per inch.

ten electronically in place. They optimistically predict newspapers delivered in this way within five years [15], despite fundamental problems yet to be overcome [19].

The advantages of a tablet are many. Unlike a desktop or notebook computer, a tablet may actually provide a pleasant reading experience. Unlike a paper publication, a tablet provides intelligent access to an unlimited amount of information; its weight is not dictated by the amount of information it carries. More than just an information appliance, it can also act as a multi-purpose multimedia communications device and interactive entertainment device.

A tablet has disadvantages too. It uses batteries which run down and have to be recharged. It may break when dropped or malfunction when exposed to hot coffee. It's not quite cheap enough to be disposable - so there's still a problem if it's misplaced or stolen. It has a "user interface" which has to be learned. The leading candidate display technology - chLCD - is still less than half as reflective (i.e. "bright") as paper.

The drawbacks of traditional paper-based publications have little to do with paper itself, and much to do with how the information gets onto the printed page. The economics of centralized printing and distribution prevent the kind of information selection, personalization and navigation people have come to expect from interactive electronic media such as the Internet. The inefficiency of printing and distributing a hundred-page newspaper to a customer who may read only a few pages is widely decried.

Given a technology such as Memjet, it becomes economic to print high-quality publications at their point of consumption rather than at their point of production. The Netpage Printer leverages Memjet to deliver personalized publications to the home, gaining many of the advantages of online publication, while retaining the ease-of-use of high-quality printed ink on paper.

Netpages and the Netpage Printer address the key problems of online publication, without relying on the development and consumer acceptance of a new reading device.

3 News and Advertising Trends

3.1 NEWSPAPERS

People obtain news from a variety of sources - network and cable television, radio, daily newspapers, and weekly newsmagazines. In the United States, although the various news media are healthy and profitable, per capita news consumption is somewhat in decline as a new generation of young adults have less time to read and favor television entertainment over news [16]. Yet six out of ten adult Americans read a newspaper every day [52].

The United States has about 1500 daily newspapers with a total circulation of 57 million. Just the top ten "national" dailies (Wall Street Journal, USA Today, New York Times, Los Angeles Times, Washington Post, etc.) account for a circulation of 10 million. The major weekly newsmagazines (Time, Newsweek, U.S. News) have a similar (weekly) circulation of about 10 million.

In 1997, newspaper companies' revenue exceeded $24 billion, a five-year high, and margins nudged 20% [38], due both to increased spending on advertising and to reduced prices of newsprint.

Television and radio, by their nature, excel at delivering breaking news. Newspapers and newsmagazines, on the other hand, deliver the depth and analysis behind the headlines. Broadcast news in isolation does a poor job of informing the public. The more local the news is, the poorer the broadcast coverage, and the greater the public's dependence on newspapers.

Newspaper content and packaging has evolved considerably since the 1970s. News is somewhat softer, news stories are shorter and more well-written, there are more feature articles, and there is more editorial and reader opinion. Newspapers are more structured. Identifiable sections make them more accessible, and provide greater focus for advertisers. Much special-interest content has migrated from daily inclusion to weekly sections. These cover topics such as lifestyle, personal finance, entertainment, technology, etc. The proportion of graphics and pictures is greater. Color is widely used. Newspapers are easier to use and more entertaining than ever before, if at the expense of some "hard" news.

Daily newspapers are growing increasingly dependent on the various wire services. A newspaper may excel at local and regional news, but rely on the major wires (Associated Press and United Press International) for national and international news, the so-called "supplemental" wires (LA Times/Washington Post, NY Times, Scripps-Howard, Knight-Ridder Tribune, etc.) for specific strengths (and value-for-money), and the international wires (Reuters etc.) for international perspective. A growing number of newspapers operate more as news aggregators than news gatherers.

Advertising typically contributes more than 75% of newspaper and magazine revenue, while subscriptions contribute less than 25% [26,50,51]. National advertising makes up roughly 14% of advertising spending, retail advertising 46%, and classified advertising 40% [51].

Advertisers are pursuing increasingly specific targeting, favoring quality newspaper readership over raw circulation [23], and using more targeted media where possible. Magazines, for example, have more specific readerships than newspapers, free "shoppers" are

very localized by their nature, while direct mailers can target demographics based on individually-categorized ZIP Codes, or databases of individuals.

Newspapers have responded with geographically zoned editions to support local advertising, and greater sectioning of their product. They have also expanded their page counts to provide more advertising scope, despite erratic newsprint prices in the 1990s [4].

Despite this, there is ongoing conflict between newspapers' mass distribution model, and advertisers' need for micro-targeting [25]. This conflict, coupled with advertisers' desire for higher-quality printing of color images, is motivating a shift from run-of-press (ROP) advertising to inserts [51]. The downside to inserts is that editorial context is lost.

3.2 ONLINE NEWS DELIVERY

Fearing the online migration of advertising, traditional news publishers from both broadcast and print have ventured into Internet-based news delivery, wanting to establish a presence at whatever cost before newcomers become entrenched. Most newspapers are still reporting losses from their online operations [51].

Online news delivery offers a number of advantages. Breaking news can be delivered as soon as it happens. News can be customized for individual readers according to their preferences and geographic locations. Readers can explore stories to arbitrary depth, follow links to related resources, and search archival material. Readers can participate in discussion groups and contribute to opinion polls. The news itself can incorporate audio and video clips, and can include live transmissions, converging with broadcast.

Online news delivery also has disadvantages. Computer screens are of limited size and quality compared with print. Few people enjoy reading a story of any length on a computer screen. Computers are not portable in the wide sense that a newspaper is. The news may be more timely, but the time and place in which it can be consumed are more constrained than with a newspaper.

Despite the power of hypertext, many online readers express a preference for a linear presentation, "where they [can] skim one section after another until the presentation [is] exhausted" [10]. Interestingly, a majority of traditional newspaper readers admit they scan every page in the main section of the newspaper [52], looking for items of interest without necessarily knowing what they're looking for, and achieving some kind of closure at the end. Online hypertext, by contrast, is both a limitless resource and a bottomless pit.

While traditional news publishers such as The New York Times can deploy full editorial content online [63], newcomers such as Yahoo typically only provide "raw" news items sourced from the wire services [47].

A recent survey indicates that 21% of the 74 million Internet users in the United States regularly read news online as an alternative to traditional print and broadcast sources, and 16% obtain a major proportion of their news online [53]. More broadly, between 37% and 64% of the Internet population reads news online at least once a week. The fluctuations in the figures are related to what may be happening in the news. Major or breaking news stories attract more users - 46% of Americans say they only follow national news stories when "something major is happening" [22].

With 41% of Americans online, the Internet population has become mainstream, and the weather has become the most popular news online. This is closely followed by technology

news, entertainment news, and local news. As one observer puts it, all of this "sound[s] like the 6 o'clock news" [53]. As a reflection of these habits, the online audience share of national newspapers has diminished from 23% in 1995 to 16% in 1998, while the online audience share of broadcast TV sites has grown.

3.3 ONLINE ADVERTISING

At its simplest, advertising alerts a motivated customer to the availability of a product, possibly at a competitive price. At a more sophisticated level, advertising seeks to influence future purchasing decisions by creating brand awareness. Ultimately, advertising seeks to create desire for a product even when actual need is absent.

Advertising prices are traditionally based on how many people see the advertisement, and their spending power in relation to the product. In practice, the more homogeneous the demographics of the audience, the easier it is to match to a product, and hence the higher the corresponding advertising cost per thousand (CPM). Broadcast media use ratings and timeslot demographics to set advertising rates. Print media use audited circulation figures and sectional readership demographics.

The simplest online advertising model is also based on how many people see the ad. Online this has the advantage of being based on solid numbers, since the number of "impressions" of a particular Web page can be counted exactly.

The specific advantage of an online ad, however, is that the ad itself can measurably capture a sales lead by acting as a link to a product site. The product site may simply provide more product information in the form of specifications, pricing, and ordering details. It may also support immediate online ordering, thus completing the link from ad to sale. Beyond providing simple ad exposure, it is this measurable linking of advertisement to sales lead or sale which is the strength of online advertising [54]. Cost per click (CLC) charging is gaining acceptance but is still controversial.

Beyond CLC, there exists the possibility of paying a commission to the ad host on any sale that actually eventuates [57]. Amazon.com is probably the best-known example of a company paying commissions to other sites in this way.

The broader advantage of online advertising is that advertising can be localized and targeted arbitrarily finely, in conjunction with the publication of online content such as news. This is the strategy pursued by online advertising agencies such as Click~Through [14], which acts as the middle-man between advertisers and online content publishers. They expect online advertising to represent more than 10 percent of all advertising revenue by 2001.

Since online ads are necessarily small-format, they communicate best with motivated customers already on the look-out for a particular product or service. Online ads are less suited to building brand awareness or creating buying desires, since the real substance of the advertisement - the product Web site - is a click away from the initial small-format ad. A small-format online ad can't provide the single-hit emotional impact of a large-format print ad, and conversely, the online world can't support the large-format ads that print can.

So-called interstitial ads, which appear full-screen when traversing from one page of information to the next, go some way to providing a medium for larger-format ads online [57]. User resistance, however, seems to be preventing their widespread use.

3.4 ONLINE CLASSIFIED ADVERTISING

Classified advertising is indisputably suited to online delivery. Unlike their traditional printed counterparts, online classifieds can be easily searched, and are not subject to space constraints. The online migration of classified advertising is considered a serious threat to newspapers' classified advertising profits [59], and some newspapers are building an online presence for this reason alone. Some observers predict as much as 50% of classified advertising revenue moving online within the next ten years [71].

Another problem faced by newspapers, who rely on classifieds for up to 40% of advertising spending, is that many newcomers are offering free online classified advertising as a way of building a venue for non-classified advertising.

4 News Personalization

From the reader's point of view, a personalized news publication can provide more information in fewer pages. The actual form this personalization takes, however, is not necessarily obvious.

The MIT Media Lab's News in the Future (NiF) project has been championing the concept of "The Daily Me" for almost two decades [44]. Nicholas Negroponte, one of the project's founders, envisages a highly personalized news publication which is no longer driven by "what other people think is news" [48]. By way of examples close to his own needs, it includes news about people and places about to be encountered, and puts "the most important [news] of all" - a summary of e-mail - on the front page [6]. Negroponte recognizes the need to vary the degree of personalization, advocating a higher "serendipity factor" on a lazy Sunday than on a working weekday.

The opposing view holds that the value of a news publication lies precisely in its *shared* nature. It reflects the common concerns and values of a community of readers, and establishes a baseline of expectations of what they are all supposed to know [66]. As a consequence, the publication also speaks with a consistent editorial voice and with consistent assumptions about the reader's level of background knowledge. Such a shared publication allows its readers to orient themselves in relation to their community.

NiF's Walter Bender answers the charge (in his own words) of "The Daily Me engendering a fragmented world populated by self-interested myopes", by stressing the possibilities of personalizing individual news items [5]. This can consist in varying the depth of an item, or supplementing it with background information, based on the reader's level of knowledge. It can involve interpreting information relative to the reader's background, such as (somewhat dubiously) making value judgements about the weather relative to the reader's normal home town weather. It can also be as simple as using metric rather than imperial units.

FishWrap [10,45], MIT's personalized campus newspaper and NiF's latest offering, goes further by creating a front page whose content represents an explicit community consensus. Each front page news item is prioritized according to the number of readers who put it forward for inclusion. The rest of the newspaper is still personalized according to each reader's profile, consisting of reader-defined sections containing topics of interest.

There are two implications of recognizing the shared nature of news. Firstly, some news is news to everybody in a community, no matter how personalized they claim they would like their news to be. This implies that the community must make decisions about news item priority, either directly (as in FishWrap) or indirectly via a proxy (i.e. an editor).

Secondly, a news item can only be properly understood in the context of the community for which it is intended. This implies that a news item must be branded with its source (assuming that the source implies the intended target). As an example, it is significant whether a news item regarding the proof of Fermat's last theorem is branded with New Scientist or The New York Times. To a professional mathematician, the latter implies, by its very existence, that the proof is of significance beyond the scientific community.

Of course, a news item must also be branded to allow its source to build and maintain its brand. The brand then allows the reader to infer the quality of the news item from the known quality of the source.

Most personalization of news uses *feature-based filtering*. This means that news item content is matched to topics and keywords in the reader's profile. News sources tag the items they produce with various information to allow them to be effectively filtered. This tagging may be brief or extensive, and may include such things as news item urgency, byline, news category, subject(s), keyword(s), date and time, and location [32]. The body text of a news item can also be scanned directly for keywords, but this may result in false matches if keywords are interpreted out of context. Items in the text such as personal names and locations can be tagged to reduce such ambiguity [32]. Similarly, dates, times, and monetary amounts can be tagged to allow localized presentation.

Feature-based filtering suffers from a number of problems. Filtering based on tags is only as good as the original tagging. The latest tagging standards are only just beginning to be adopted [7]. Filtering based on the text itself is constrained by the intelligence of the text parsing. If based simply on keyword matching, it can be both inaccurate, generating false matches because of word sense ambiguity, and imprecise, generating false mismatches because of a lack of inference.

Feature-based filtering is incapable of discerning more abstract attributes such as quality, style, and point-of-view (unless they're indicated by tags). And since it only matches items anticipated by the user's profile, it is a poor generator of serendipitous finds.

FishWrap's front page comes into existence based on a crude form of *collaborative filtering*. In its broader form, collaborative filtering involves sharing recommendations (or ratings) among *like-minded* people [55]. This means that one person's ratings influence another person if and only if the two share similar interests, i.e. they have similar rating histories. Collaborative filtering overcomes many of the problems of feature-based filtering, since ratings originate with people who have digested the items in question, rather than from automated analysis of the items. Collaborative filtering sidesteps the issue of *explaining* why a person might like a particular item.

Collaborative filtering has problems of its own. The system only works if people are willing to contribute ratings. In contributing ratings, of course, they are both doing the community a service and tuning their own interest profiles. The statistical error in correlating people's interests decreases as the number of ratings increases. However, incentives may have to be offered to encourage people to contribute ratings.

To bootstrap the accumulation of ratings for new items, an independent mechanism must exist to distribute them to a critical mass of people. Conversely, to bootstrap the accumulation of interest profiles for new users, an independent mechanism must exist to distribute a critical mass of items to them.

To allow meaningful accumulation of ratings, a sufficient period of time must be allowed to elapse. This may conflict with the timely delivery of items in question.

The statistical correlation between different people's interests, represented by their rating histories, is most meaningful when the ratings apply to homogeneous items. For a set of heterogeneous items, collaborative filtering is best applied to homogeneous subsets.

In a news setting, collaborative filtering is best applied to feature articles. Features have the longer life span required to support the accumulation of ratings, and are often appreciated for abstract qualities best singled out by collaborative filtering (good writing, humor, incisiveness, etc.).

Naturally, the larger a publication's readership, and the better its taste in relation to a potential reader, the stronger the publication's brand will appear to that reader.

Although it's easy to become preoccupied with automatic filtering, in reality there's more to editing the news than just filtering news feeds. An editor also solicits news, commissions analysis, and offers opinion, ideally ensuring that the publication offers a balanced and complete view of the world.

Perhaps the most important personalization step a reader takes is in selecting a particular publication from a set of available publications, based on its perceived quality and relevance.

Thus the publication's brand equates to the highest-level and most useful filter of all.

ARCHITECTURE

5 Netpage System Architecture

5.1 THE INTERNET

The Internet is a worldwide collection of interconnected networks which communicate using the TCP/IP protocol suite [58]. A TCP/IP-based internetwork not connected to *the* Internet is often referred to as *an* internet (i.e. with a lower-case 'i'). When an internet is deployed within an organization, it is often termed an intranet.

Access to the Internet is widespread in developed countries. In the United States, for example, 41% of the population has access to the Internet [53].

While most consumers still access the Internet via low-speed dial-up modems connected to the switched telephone system, inexpensive broadband access is becoming available to a growing number of households via the cable networks. Cable service is available to 95% of American households [56], and cable modem service is available to a 20% subset [8]. While dial-up modems offer speeds of up to 56Kbps, cable modems offer practical speeds of up to about 3Mbps[1], i.e. over 50 times faster.

DSL (Digital Subscriber Line) [11,12], while offering similar speeds to cable modems but via the telephone system, is not yet widely used. ISDN (Integrated Services Digital Network), although widely used for corporate access, has had little consumer impact due to its high price and comparatively low performance.

The deployment of third-generation (3G) cellular telephony within the next few years will bring practical mobile broadband speeds of 2Mbps [21]. 3G cellular uses WCDMA (wideband code-division multiple access), a spread-spectrum technology. Satellite systems are arguably closer to offering even faster broadband Internet access [28,49,62].

The core of the Internet is made up of a number of independent high-speed fiber-optic networks connected into NAPs (Network Access Points) or peered directly. These have until recently used single-wavelength TDM (Time-Division Multiplexing) SONET (Synchronous Optical Network) transmission systems which utilize about 1% of an optic fiber's capacity to yield a 2.5Gbps OC-48 channel[2]. Carriers are now beginning to deploy multi-wavelength DWDM (Dense Wavelength-Division Multiplexing) systems which yield up to 40 such channels per optic fiber, thus increasing network capacity significantly without requiring the laying of more fiber [9,39]. Internet architects are therefore now contemplating aggregate capacity in the terabit (Tbps) range.

The Internet uses the four-layer TCP/IP protocol suite. The application layer provides various end-to-end application services, and is a client of the transport layer which provides end-to-end delivery services. The transport layer in turn is a client of the network layer which provides packet routing. The network layer is a client of the link layer which encapsulates specifics of the protocols and hardware of the actual communications links.

The core Internet transport protocol, TCP (Transmission Control Protocol), provides a reliable end-to-end delivery service. The core Internet network protocol, IP (Internet Protocol), provides an unreliable and connectionless packet routing service. IP may lose or

1. Although the cable supports 30Mbps and the cable modem theoretically supports 10Mbps.
2. SONETchannels have an OC-*n* designation, where OC stands for Optical Carrier, and *n* gives the channel speed in units of about 52Mbps. An OC-48 channel therefore has a speed of about 2.5Gbps.

deliberately discard packets, and may deliver packets out of order, and it is the responsibility of a higher layer to provide a reliable end-to-end service.

With the proliferation of streaming media services on the Internet, support for multicast is spreading rapidly. Multicast is a form of broadcast with a specific set of recipients. It makes efficient use of network capacity because a packet traverses a network link once rather than once per recipient. It is particularly efficient if the recipients are connected to the Internet via an intrinsically broadcast medium such as cable or satellite. The @Home cable network has successfully enabled multicasting of streaming media services [37].

IP Multicast is an extension of IP, and so is unreliable. While this is often acceptable for time-critical data such as streaming video, it may not be acceptable for other shared data types. Significant effort is being expended to develop reliable multicast transport protocols on top of IP Multicast. Although several reliable multicast protocols are available and have been deployed [40,41,29,30], the Internet standardization process is incomplete [31].

5.2 NETPAGES AND NETPAGE DOCUMENTS

Netpages are the foundation on which a Netpage Network is built. They provide a paper-based user interface to published information and interactive services.

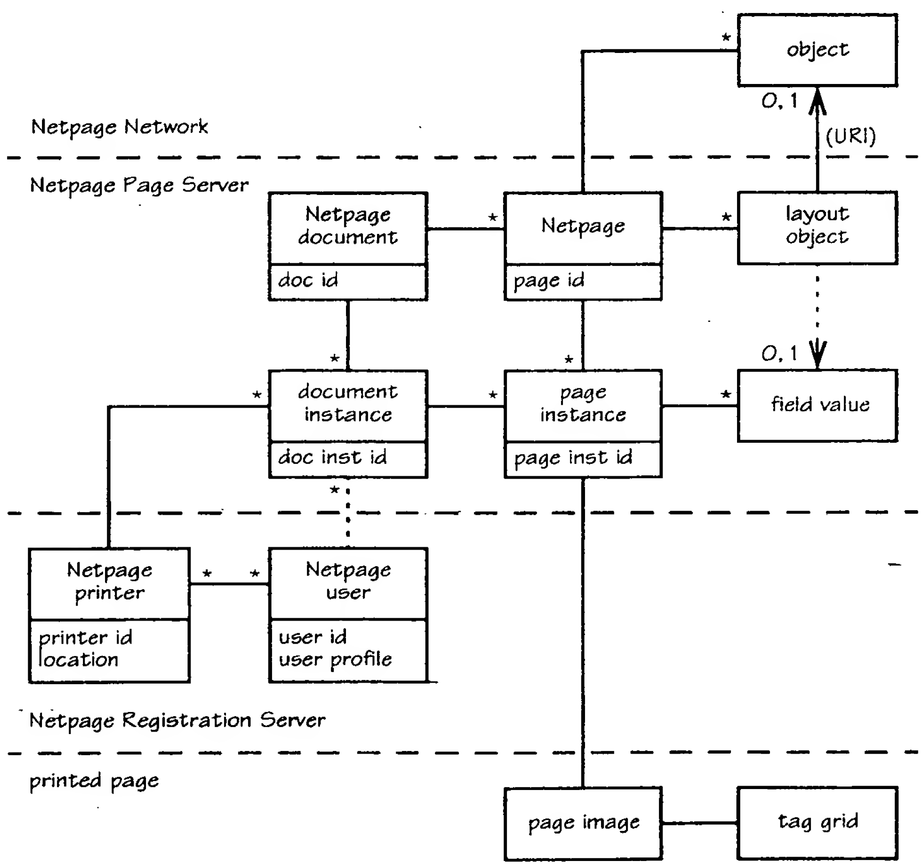


Figure 1. Netpage Document structure (* indicates an n-ary relationship)

Each Netpage consists of a compact page layout maintained persistently by a Netpage Page Server. The page layout refers to objects such as images, fonts and pieces of text, typically stored elsewhere on the Netpage Network.

Netpages are organized into Netpage Documents. Both Netpages and Netpage Documents are assigned globally unique identifiers.

Each Netpage Document has a set of document instances, each of which describes a printed instance of the document. Each Netpage in the Netpage Document has a corresponding set of page instances, each of which describes a printed instance of the page. Both page instances and document instances are assigned globally unique identifiers. They are also uniquely associated with the printer on which they are printed and the user who initiated the print request, if known.

Each page instance maintains a set of user-supplied values for fields in the page layout. This ensures that user input is captured and stored independently for each page instance. The separation of page instances and Netpages is crucial for pages which contain input fields, i.e. forms. It is not crucial for pages devoid of input fields, but still useful because it supports independent auditing of each page instance.

The physical page image includes encoded information which identifies the page instance and hence the Netpage to which it corresponds. It also includes encoded information which superimposes an addressable spatial grid over the page image, to allow pen actions performed relative to the page image to be correlated with the contents of the page layout.

The encoded information is normally printed in infrared-absorptive ink on any normal paper substrate which is infrared-reflective. Near-infrared wavelengths are invisible to the human eye but are easily sensed by a solid-state image sensor with an appropriate filter.

The encoded information is picked up by an infrared-imaging pen and transmitted to the associated Netpage Printer. The pen is wireless and communicates with the Netpage Printer using an encrypted radio frequency signal.

The encoded information is organized as a set of tags, each containing both the id of the page instance and the position of the tag. The tags tile the entire page image, and are sufficiently small and densely arranged that the pen can reliably image at least one tag even on a single click on the page. It is important that the pen recognize the page instance id and position on every interaction with the page, since the interaction is stateless.

The tags are error-correctably encoded to make them resilient to errors introduced by dirt on the page or during the imaging process.

Memjet-based Netpage Printers are designed to print a tag grid using infrared (IR) ink. Printers not enabled for IR printing have the option to print tags using IR-absorptive black ink, although this restricts tags to otherwise empty areas of the page. Although such pages have more limited functionality than IR-printed pages, they are still classed as Netpages.

5.3 THE NETPAGE NETWORK

A Netpage Network consists of a distributed set of Netpage Publication Servers, Netpage Page Servers, and Netpage Printers connected via an internet. In technological terms this document describes *any* Netpage Network. In business terms it usually refers to *the* Netpage Network connected via *the* Internet.

As described above, a Netpage Page Server maintains persistent information about Netpage Documents, Netpages, and their printed instances, to allow pen operations on printed pages to be interpreted intelligently.

The Netpage Network includes any number of Netpage Page Servers, each handling a subset of Netpages. As described above, each page instance is identified by a globally unique id which is encoded in the tag grid of the corresponding printed page. The Netpage Printer uses this id to retrieve the page layout of the page from a Netpage Page Server when it needs to interpret pen operations relative to the page.

The Netpage Printer uses the internet Distributed Name System (DNS) to resolve a Netpage instance id into a page instance maintained by a particular Netpage Page Server.

The DNS is a protocol and a hierarchical system of name servers used to resolve internet domain names into resources. Planned enhancements to the DNS allow it to be used to resolve more general Uniform Resource Identifiers (URIs), and in particular Uniform Resource Names (URNs), into resource locations [18]. Netpage instance ids are formulated as URNs, allowing the enhanced DNS to be used to resolve them. In the absence of timely standardization and deployment of an enhanced DNS on the Internet, the Netpage Network can deploy its own system of enhanced name servers.

A Netpage Publication Server is an internet server which publishes Netpage Documents to Netpage Printers. It is described in Section 6.

5.4 THE NETPAGE PRINTER

The Netpage Printer is the appliance which prints Netpage Documents. It is connected to a Netpage Network via an internet, ideally via a broadband connection.

Apart from identity and security settings in non-volatile memory, the Netpage Printer contains no persistent storage. As far as a user is concerned, *the network is the computer* [60]. Netpages function interactively across space and time with the help of the distributed Netpage Page Servers, independently of particular Netpage Printers.

The Netpage Printer receives Netpage Documents from Netpage Publication Servers. Each document is distributed in two parts: the page layouts, and the actual text and image objects which populate the pages. Because of personalization, page layouts are typically specific to a particular subscriber and so are pointcast to the subscriber's printer. Text and image objects, on the other hand, are typically shared with other subscribers, and so are multicast to all subscribers' printers.

The Netpage Publication Server optimizes the segmentation of document content into pointcasts and multicasts. After receiving the pointcast of a document's page layouts, the printer knows which multicasts, if any, to listen to.

Once the printer has received the entire document's page descriptions, i.e. page layouts *and* objects, it can print the document.

The printer rasterizes and prints odd and even pages simultaneously on both sides of the sheet. It therefore contains duplexed print engines and imaging units.

The printing process consists of two decoupled stages: rasterization of page descriptions, and expansion and printing of page images. The raster image processor (RIP) consists of

one or more standard DSPs running in parallel. The duplexed print engines consist of custom processors which expand, dither and print page images in real time, synchronized with the operation of the printheads in the imaging units.

There are four major design variations embodied in the various Netpage Printer models:

- *form factor*: pocket, portable, desktop, wall-mount or tabletop
- *printhead width*: 4" (photo), 8½" (portrait Letter) or 11" (landscape Letter)
- *paper source*: cut sheet or print cartridge
- *Internet connection*: wired or wireless

The form factor variations yield five basic models, each with variants determined by printhead width (and hence printing speed), and paper source. Eight planned models are defined in Table 1, and illustrated in Figure 2.

Table 1. Netpage Printer models

model	form factor	variant	printhead width	paper source	Internet connect
Microprinter	pocket	R	4"	cartridge	wireless
Travelprinter	portable	R	8½"	cartridge	wireless
Deskprinter	desktop	R		cartridge	wired or wireless
Wallprinter	wall-mount	-		cut sheet	
		Pro	11"	cut sheet	
		Pro R		cartridge	
Tableprinter	tabletop	Pro		cut sheet	
		Pro R		cartridge	

The Deskprinter, Wallprinter and Tableprinter models can be factory-configured with various network modules, allowing both wired and wireless versions. The Microprinter and Travelprinter both use a cellular telephone module, with the promise of broadband speed within a few years.

The Wallprinter models are ideal for unobtrusive installation in a home, while the Tableprinter models might be preferred in an office environment. Note that the Tableprinter models are Wallprinter models factory-adapted for tabletop use via a stand. The Deskprinter, with its small footprint, is ideal for both home and office use.

The Microprinter prints normal Netpages at quarter size, and provides full wireless Netpage Network access in a pocket device.

The paper roll cartridge contains both paper and ink. The paper is in the form of a continuous roll, cut on demand by the printer. The 11" print cartridge has a capacity of 1000 Letter sheets. It also contains the glue supply for binding the sheets of a document together. The 8½" print cartridge has a capacity of 50 Letter sheets, or equivalently 100 A5 sheets. The 4" print cartridge has a capacity of 36 6×4 photos, or 41 quarter-size Netpages. The 8½" and 4" print cartridges don't contain a glue supply because neither the Microprinter nor the Travelprinter includes a binding mechanism.

The 4" printhead models print at 30 quarter-size pages per minute. The 8½" printhead models print at 60 pages per minute, or 30 duplex sheets per minute. The 11" printhead models print at 90 pages per minute, or 45 duplex sheets per minute.

Figure 2. Netpage Printer family

5.5 THE NETPAGE PEN

The Netpage Pen operates both as a normal marking ink pen and as a non-marking stylus. When either nib is in contact with a Netpage, the pen continuously monitors its movements relative to the page. The nib is attached to a pressure sensor. The pen pressure can be interpreted relative to a threshold to indicate whether the pen is "up" or "down". It can also be interpreted as a continuous value, for example when the pen is capturing a signature, to allow the full dynamics of the signature to be verified.

The pen determines the position of its nib on the Netpage by imaging, in the infrared spectrum, an area of the page in the vicinity of the nib. It decodes the nearest page id and position tag, and adjusts the position given by the tag to account for the distance between the area imaged and the actual nib, and the position of the tag in the imaged area. Although the position resolution of the tag may be low, because the tag density on the page is inversely proportional to the tag size, the adjusted position resolution is quite high, and easily exceeds the minimum 200 dpi resolution required for handwriting recognition [61].

Pen actions relative to a Netpage consist of a series of strokes. A stroke consists of a sequence of time-stamped pen positions on the page, initiated by a pen-down event and completed by the subsequent pen-up event. A stroke is also tagged with the page id of the Netpage whenever the page id changes, i.e. just at the start of the stroke under normal circumstances.

The position tags on the Netpage contain various control bits. One of these instructs the pen to activate its "active area" LED. Thus a region on the page which corresponds to the active area of a button or hyperlink can be encoded to activate this LED, giving the user visual feedback that the button or hyperlink is active when the pen passes over it. Another control bit instructs the pen to capture continuous pen pressure readings and tag the stroke with these readings. Thus a region on the page which corresponds to a signature input area can be encoded to capture continuous pen pressure.

Whenever the pen is within range of a printer with which it can communicate, the pen slowly flashes its "online" LED. When the pen fails to decode a stroke relative to the page, it momentarily activates its "error" LED. When the pen succeeds in decoding a stroke relative to the page, it momentarily activates its "ok" LED.

The pen also contains a pair of passive accelerometers mounted at right angles to each other in the plane normal to the pen's axis. The accelerometers respond to gravity and allow the pen to compute its tilt. This in turn helps it auto-focus its optics and compute the nib-to-tag displacement. If the stroke is being tagged with pen pressure readings, then it is also tagged with tilt readings.

A sequence of captured strokes, whether tagged with pen pressure and tilt or not, is referred to as *digital ink*. Digital ink forms the basis for the digital exchange of drawings and handwriting, for on-line recognition of handwriting [61], and for on-line verification of signatures.

The pen is wireless and transmits digital ink to the Netpage Printer using a radio frequency signal. The digital ink data is encrypted for security and packetized for efficient transmission, but is always flushed on a pen-up event to ensure timely handling in the printer.

When the pen is out-of-range of a printer it buffers digital ink in internal memory, which has a capacity of more than 12 minutes of continuous handwriting. When the pen is once again within range of a printer, it transfers any buffered digital ink.

A pen can be registered with any number of printers, but because all state data resides in Netpages both on paper and on the network, it is largely immaterial which printer a pen is communicating with at any particular time.

5.6 NETPAGE INTERACTION

When the Netpage Printer receives a digital ink stroke from the pen, it retrieves the page layout of the Netpage identified in the stroke, to allow it to correctly interpret the stroke. The printer resolves, via the DNS, the address of the Netpage Page Server which holds the page layout, and then retrieves the page layout from the server. If the page was recently identified in an earlier stroke, then the printer may already have the address of the relevant Netpage Page Server in its cache. It may also have the page layout itself in its cache, in which case there may be no need to retrieve it.

Once the printer has the page layout of the Netpage to which the pen stroke refers, it can interpret the stroke in relation to the layout and content of the page. This involves hit-testing the objects on the page to determine which objects the pen is interacting with, in much the same way that mouse movements and button presses are interpreted in a graphical user interface system.

A "click" is a stroke where the distance between the pen down position and the subsequent pen up position is less than some small maximum. An object which is activated by a click requires a click to be activated, i.e. a longer stroke is ignored. The failure of a pen action, such as a "sloppy" click, to register is indicated by the lack of response from the pen's "ok" LED.

There are two kinds of interactive objects on a Netpage: hyperlinks and form fields.

When a hyperlink is activated, the printer sends a request to a handler somewhere on the network. The handler is identified by a URI, and the URI is resolved in the normal way via the DNS. There are three types of hyperlinks: general hyperlinks, form hyperlinks, and selection hyperlinks. A general hyperlink may implement a request for a linked document, or may simply signal a preference to a server. A form hyperlink submits the corresponding form to a form handler. A selection hyperlink submits the current selection to a selection handler. If the current selection contains a single-word piece of text, for example, the selection handler may return a single-page document giving the word's meaning within the context in which it appears, or a translation into a different language. Each hyperlink type is characterized by what information is submitted to the handler.

Form fields come in four varieties: checkboxes, text areas, digital ink areas, and signature areas. A checkbox accepts a true or false value. Any mark (a tick, a cross, a stroke, a fill zigzag, etc.) captured in a checkbox area is assigned as a true value to the corresponding field. A text area accepts a text string. Any digital ink captured in a text area is automatically converted to text via on-line handwriting recognition and the text is assigned to the corresponding field. A digital ink area accepts raw digital ink. Any digital ink captured in a digital ink area is assigned to the corresponding field. A signature area accepts a hand-written signature. Any digital ink captured in a signature area is automatically verified and the resulting signature token is assigned to the corresponding signature field. Signature verification is discussed in more detail in Section 8.

"Editing" commands, such as strike-throughs indicating deletion, are also recognized in form fields.

Table 2. Summary of pen interactions with a Netpage

object	type	pen input	action
hyperlink	general	click	submit action to handler via URI
	form	click	submit form to handler via URI
	selection	click	submit selection to handler via URI
form field	checkbox	any mark	set field value to true
	text area	handwriting	convert digital ink to text; assign text to field
	digital ink area	digital ink	assign digital ink to field
	signature area	signature	verify digital ink signature; assign signature token to field
none	-	circumscription	convert digital ink to region; select object(s) in region

Because the handwriting recognition algorithm works "on-line" (i.e. with access to the dynamics of the pen movement), rather than "off-line" (i.e. with access only to a bitmap of pen markings), it can recognize run-on discretely-written characters [61] with high accuracy, without a writer-dependent training phase.

Digital ink, as already stated, consists of a sequence of strokes. Any stroke which starts in a particular object's active area is appended to that area's digital ink stream, ready for eventual interpretation. Any stroke not appended to an object's digital ink stream is appended to the remaining inactive area's digital ink stream.

Digital ink captured in the inactive area is interpreted as a selection gesture. Any circumscription of one or more objects is interpreted as a selection of the circumscribed objects.

The printer maintains a current selection for each pen. The selection contains the most recent object selected, resolved with reference to the page layout and content. The selection can be attached to or pasted into another form, or in general be submitted to a selection handler as described earlier. The selection is cleared after an inactivity time-out to ensure predictable behavior.

Table 2 provides a summary of pen interactions with a Netpage.

5.7 FORMS

As described in Section 5.2, user input on a physical Netpage is ultimately recorded persistently by a Netpage Page Server together with the corresponding page instance. To ensure efficient capture of user input, the printer accumulates input locally. To prevent update anomalies, however, the printer temporarily obtains exclusive access to the page instance from the Netpage Page Server. The printer flushes input back to the server and relinquishes exclusive access when the user initiates a non-local action on the page; after an inactivity time-out on the page; when the printer wishes to free up local storage consumed by the page; and on request from the server.

When the printer submits a form to a form handler, it simply submits the document instance of the form. The form handler retrieves the field values from the Netpage Page Server at its leisure.

A form can also act as a shared "blackboard" between the user and the form handler, i.e. the form handler can query the contents of the form fields maintained by the Netpage Page Server without the user explicitly submitting the form.

For text areas, the raw digital ink is optionally also stored with the page instance on the Netpage Page Server. This allows the form handler to interrogate the raw digital ink should it suspect the original recognition of the handwriting. This might involve human intervention at the application level for forms which fail certain application-specific consistency checks. As an extension to this, the entire background area of a form can be designated as a digital ink area. The form handler can then decide, on the basis of the presence of digital ink outside the explicit fields of the form, to route the form to a human operator, on the assumption that the user may have indicated amendments to the filled-in fields outside of those fields.

Form fields can optionally be tagged to indicate their meaning. Fields tagged in this way may include name and address fields, for example. This semantic tagging allows these fields to be automatically filled in whenever a "blank" form is requested by an identifiable user, i.e. a user who has registered their identity with the system and linked it to the identity of their pen.

5.8 Standard Features of Netpages

Each Netpage is printed with the Netpage logo at the bottom to indicate that it is a Netpage and therefore has interactive properties. The logo also acts as a "copy" button. In most cases pressing the logo produces a copy of the page. In the case of a form the button instead elicits a page giving the user the option to print the entire form document. And in the case of a secure document, such as a ticket or coupon, the button elicits an explanatory note or advertising page.

The default single-page copy function is handled directly by the relevant Netpage Page Server. Special copy functions are handled by linking the logo button to other URIs.

Once a Netpage form has been submitted, it is marked as submitted by the Netpage Page Server and cannot be submitted again. An attempt to do so elicits a status report indicating when it was submitted. A copy of the form can still be made, altered, and re-submitted.

5.9 The Help System

The Netpage Printer has a single button labelled "help". When pressed it elicits a single page of information. This information includes the following:

- status of printer connection
- status of printer consumables
- top-level help menu
- document function menu
- top-level Netpage Network directory

The help menu provides a hierarchical manual on how to use the Netpage System.

The document function menu includes the following functions:

- print a copy of a document

- print a clean copy of a form
- print the status of a document

A document function is initiated by simply pressing the button and then touching any page of the document. The status of a document indicates who published it and when, to whom it was delivered, and to whom and when it was subsequently submitted as a form.

The Netpage Network directory allows the user to navigate the hierarchy of publications and services on the network. As an alternative, the user can call the Netpage Network "900" number "yellow pages" and speak to a human operator. The operator can locate the desired document and route it to the user's printer. Depending on the document type, the publisher or the user pays the small "yellow pages" service fee.

The help page is obviously unavailable if the printer is unable to print. In this case the "error" light is lit and the user can request remote diagnosis over the network.

6 Personalized Publication Model

In the following discussion, news is used as a canonical publication example to illustrate personalization mechanisms in the Netpage System. Although news is often used in the limited sense of newspaper and newsmagazine news, the intended scope is wider.

In the Netpage System, the editorial content and the advertising content of a news publication are personalized using different mechanisms. The editorial content is personalized according to the reader's explicitly stated and implicitly captured interest profile. The advertising content is personalized according to the reader's locality and demographic.

6.1 Editorial Personalization

A subscriber can draw on two kinds of news sources: those that deliver news publications, and those that deliver news streams. While news publications are aggregated and edited by the publisher, news streams are aggregated either by a news publisher or by a specialized news aggregator. News publications typically correspond to traditional newspapers and newsmagazines, while news streams can be many and varied: a "raw" news feed from a news service, a cartoon strip, a freelance writer's column, a friend's bulletin board, or the reader's own e-mail.

The Netpage Publication Server supports the publication of edited news publications as well as the aggregation of multiple news streams. By handling the aggregation and hence the formatting of news streams selected directly by the reader, the server is able to place advertising on pages over which it otherwise has no editorial control.

The subscriber builds a daily newspaper by selecting one or more contributing news publications, and creating a personalized version of each. The resulting daily editions are printed and bound together into a single newspaper. The various members of a household typically express their different interests and tastes by selecting different daily publications and then customizing them.

For each publication, the reader optionally selects specific sections. Some sections appear daily, while others appear weekly. The daily sections available from The New York Times online, for example, include "Page One Plus", "National", "International", "Opinion", "Business", "Arts/Living", "Technology", and "Sports". The set of available sections is obviously specific to a publication, as is the default subset.

The reader extends the daily newspaper by creating custom sections, each one drawing on any number of news streams. Custom sections might be created for e-mail and friends' announcements ("Personal"), or for monitoring news feeds for specific topics ("Alerts" or "Clippings").

For each section, the reader optionally specifies its size, either qualitatively (e.g. short, medium, or long), or numerically (i.e. as a limit on its number of pages), and the desired proportion of advertising, either qualitatively (e.g. high, normal, low, none), or numerically (i.e. as a percentage).

The reader also optionally expresses a preference for a large number of shorter articles or a small number of longer articles. Each article is ideally written (or edited) in both short and long forms to support this preference.

An article may also be written (or edited) in different versions to match the expected sophistication of the reader, for example to provide children's and adults' versions. The appropriate version is selected according to the reader's age. The reader can specify a "reading age" which takes precedence over their biological age.

The articles which make up each section are selected and prioritized by the editors, and each is assigned a useful lifetime. By default they are delivered to all relevant subscribers, in priority order, subject to space constraints in the subscribers' editions.

In sections where it is appropriate, the reader may optionally enable collaborative filtering. This is then applied to articles which have a sufficiently long lifetime. Each article which qualifies for collaborative filtering is printed with rating buttons at the end of the article. The buttons can provide an easy choice (e.g. "liked" and "disliked'), making it more likely that readers will bother to rate the article.

Articles with high priorities and short lifetimes are therefore effectively considered essential reading by the editors and are delivered to most relevant subscribers.

The reader optionally specifies a serendipity factor, either qualitatively (e.g. do or don't surprise me), or numerically. A high serendipity factor lowers the threshold used for matching during collaborative filtering. A high factor makes it more likely that the corresponding section will be filled to the reader's specified capacity. A different serendipity factor can be specified for different days of the week.

The reader also optionally specifies topics of particular interest within a section, and this modifies the priorities assigned by the editors.

The speed of the reader's Internet connection affects the quality at which images can be delivered. The reader optionally specifies a preference for fewer images or smaller images or both. If the number or size of images is not reduced, then images may be delivered at lower quality (i.e. at lower resolution or with greater compression).

At a global level, the reader specifies how quantities, dates, times and monetary values are localized. This involves specifying whether units are imperial or metric, a local timezone and time format, and a local currency, and whether the localization consist of *in situ* translation or annotation. These preferences are derived from the reader's locality by default.

To reduce reading difficulties caused by poor eyesight, the reader optionally specifies a global preference for a larger presentation. Both text and images are scaled accordingly, and less information is accommodated on each page.

The language in which a news publication is published, and its corresponding text encoding, is a property of the publication and not a preference expressed by the user. However, the Netpage Network may provide automatic translation services in various guises.

6.2 ADVERTISING LOCALIZATION AND TARGETING

The personalization of the editorial content directly affects the advertising content, because advertising is typically placed to exploit the editorial context. Travel ads, for example, are more likely to appear in a travel section than elsewhere. The value of the editorial content to an advertiser (and therefore to the publisher) lies in its ability to attract large numbers of readers with the right demographics.

Effective advertising is placed on the basis of locality and demographics. Locality determines proximity to particular services, retailers etc., and particular interests and concerns associated with the local community and environment. Demographics determine general interests and preoccupations as well as likely spending patterns.

A news publisher's most profitable product is advertising "space", a multi-dimensional entity determined by the publication's geographic coverage, the size of its readership, its readership demographics, and the page area available for advertising.

In the Netpage System, the Netpage Publication Server computes the approximate multi-dimensional size of a publication's saleable advertising space on a per-section basis, taking into account the publication's geographic coverage, the section's readership, the size of each reader's section edition, each reader's advertising proportion, and each reader's demographic.

In comparison with other media, the Netpage System allows the advertising space to be defined in greater detail, and allows smaller pieces of it to be sold separately. It therefore allows it to be sold at closer to its true value.

For example, the same advertising "slot" can be sold in varying proportions to several advertisers, with individual readers' pages randomly receiving the advertisement of one advertiser or another, overall preserving the proportion of space sold to each advertiser.

The Netpage System allows advertising to be linked directly to detailed product information and online purchasing. It therefore raises the intrinsic value of the advertising space.

Because personalization and localization are handled automatically by Netpage Publication Servers, an advertising aggregator can provide arbitrarily broad coverage of both geography and demographics. The subsequent disaggregation is efficient because it is automatic. This makes it more cost-effective for publishers to deal with advertising aggregators than to directly capture advertising. Even though the advertising aggregator is taking a proportion of advertising revenue, publishers may find the change profit-neutral because of the greater efficiency of aggregation. The advertising aggregator acts as an intermediary between advertisers and publishers, and may place the same advertisement in multiple publications.

It is worth noting that ad placement in a Netpage publication can be more complex than ad placement in the publication's traditional counterpart, because the publication's advertising space is more complex. While ignoring the full complexities of negotiations between advertisers, advertising aggregators and publishers, it is clear that the Netpage System should ideally provide some automated support for these negotiations, including support for automated auctions of advertising space. Automation is particularly desirable for the placement of advertisements which generate small amounts of income, i.e. small or highly localized advertisements.

Once placement has been negotiated, the aggregator captures and edits the advertisement and records it on a Netpage Ad Server. Correspondingly, the publisher records the ad placement on the relevant Netpage Publication Server. When the Netpage Publication Server lays out each user's personalized publication, it picks the relevant advertisements from the Netpage Ad Server.

6.3 USER PROFILES

The personalization of news and other publications relies on an assortment of user-specific profile information:

- publication customizations
- collaborative filtering vectors
- contact details
- presentation preferences

The customization of a publication is typically publication-specific, and so the customization information is maintained by the relevant Netpage Publication Server.

A collaborative filtering vector consists of the user's ratings of a number of news items. As described in Section 4, it is used to correlate different users' interests for the purposes of making recommendations. Although there are benefits to maintaining a single collaborative filtering vector independently of any particular publication, there are two reasons why it is more practical to maintain a separate vector for each publication: there is likely to be more overlap between the vectors of subscribers to the same publication than between those of subscribers to different publications; and a publication is likely to want to present its users' collaborative filtering vectors as part of the value of its brand, not to be found elsewhere. Collaborative filtering vectors are therefore also maintained by the relevant Netpage Publication Server.

Contact details, including name, street address, ZIP Code, state, country, telephone numbers, etc., are by their nature global and are maintained by a Netpage Registration Server.

Presentation preferences, including those for quantities, dates and times discussed in Section 6.1, are likewise global and maintained in the same way.

The localization of advertising relies on the locality indicated in the user's contact details, while the targeting of advertising relies on personal information such as date of birth, gender, marital status, income, profession, education, etc., or qualitative derivatives such as age range and income range.

For those users who choose to reveal personal information for advertising purposes, the information is maintained by the relevant Netpage Registration Server. In the absence of such information, advertising can be targeted on the basis of the demographic associated with the user's ZIP or ZIP+4 Code.

Each user, pen, printer, publisher and publication is assigned its own globally unique identifier, and the Netpage Registration Server maintains the relationships between them.

Each user may be authorized to use any number of printers, and each printer may allow any number of users to use it. Each user has a single default printer, to which periodical publications are delivered. The server keeps track of which publishers a user has authorized print to the user's default printer.

Each user may have several pens, but a pen is specific to a single user. If a user is authorized to use a particular printer, then that printer recognizes any of the user's pens.

These relationships are illustrated in Figure 3.

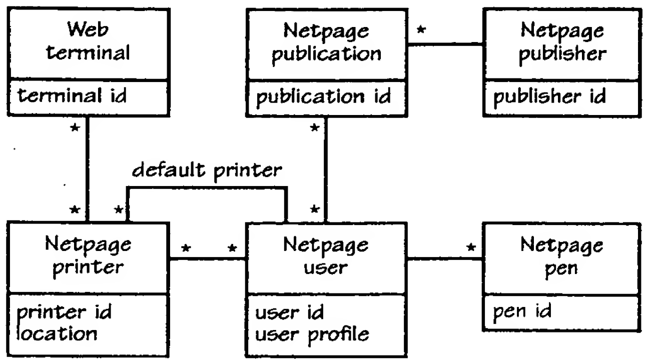


Figure 3. User registration relationships

The pen identifier is used, in the form of a URN, to locate the corresponding user profile maintained by a particular Netpage Registration Server, via the DNS in the usual way.

A Web terminal can be authorized to print on a particular Netpage Printer, allowing Web pages and Netpage documents encountered during browsing to be conveniently printed on the nearest Netpage Printer.

6.4 INTELLIGENT PAGE LAYOUT

The Netpage Publication Server automatically lays out the pages of each user's personalized publication on a section-by-section basis. Since most advertisements are in the form of pre-formatted rectangles, they are placed on the page before the editorial content.

The advertising ratio for a section can be achieved with wildly varying advertising ratios on individual pages within the section, and the ad layout algorithm exploits this. The algorithm attempts to co-locate closely tied editorial and advertising content, e.g. ads for roofing material placed specifically with the publication because of a special feature on do-it-yourself roofing repairs.

The editorial content selected for the user, i.e. text with associated images and graphics, is then laid out according to various aesthetic rules.

The entire process, including the selection of ads and the selection of editorial content, must be iterated once the layout has converged, to attempt to more closely achieve the user's stated section size preference. The section size preference can, however, be matched *on average* over time, allowing significant day-to-day variations.

6.5 DOCUMENT FORMAT

Once the document is laid out, it is encoded for efficient distribution and persistent storage on the Netpage Network.

The primary efficiency mechanism is the separation of information specific to a single user's edition and information shared between multiple users' editions. The specific information consists of the page layout. The shared information consists of the objects to which the page layout refers, including images, graphics, and pieces of text.

A text object contains fully-formatted text represented in the Extensible Markup Language (XML) [67] using the Extensible Stylesheet Language (XSL) [68]. XSL provides precise control over text formatting independently of the region into which the text is being set, which in this case is being provided by the layout. The text object contains embedded language codes to enable automatic translation, and embedded hyphenation hints to aid with paragraph formatting.

An image object encodes an image in the JPEG 2000 wavelet-based compressed image format [33]. The original DCT-based JPEG algorithm introduces negligible visual loss at compression ratios below 10:1 [64]. JPEG 2000 is planned to achieve the same quality at compression ratios 30% higher, i.e. at about 13:1 [34].

A graphic object encodes a 2D graphic in Scalable Vector Graphics (SVG) [70] format.

The layout itself consists of a series of placed image and graphic objects, linked textflow objects through which text objects flow, hyperlinks and input fields as described in Section 5.6, and watermark regions. These layout objects are summarized in Table 3. The layout uses a compact format suitable for efficient distribution and storage.

The layout is tagged with the version of the text-setting algorithm used by the Netpage Publication Server when the layout was first created, allowing the Netpage Printer to exactly reproduce the physical layout intended by the server.

Because Netpage Printer software is automatically upgraded over the Netpage Network, it is feasible for Netpage Printers to contain every version of the text-setting algorithm.

Table 3. Netpage layout objects

layout object	attribute	format of linked object
image	position	-
	image object URI	JPEG 2000
graphic	position	-
	graphic object URI	SVG
textflow	textflow id	-
	region[a]	-
	optional text object URI	XML/XSL
hyperlink	type	-
	region[a]	-
	handler URI	-
field	type	-
	meaning	-
	region[a]	-
watermark	region[a]	-

a. arbitrary multi-edged shape defined with spline paths

6.6 Document Distribution

As described above, for purposes of efficient distribution and persistent storage on the Netpage Network, a user-specific page layout is separated from the shared objects to which it refers.

When a subscribed publication is ready to be distributed, the Netpage Publication Server allocates, with the help of the Netpage Id Server, a globally unique id for each page, page instance, document, and document instance.

The server computes a set of optimized subsets of the shared content and creates a multicast channel for each subset, and then tags each user-specific layout with the names of the multicast channels which will carry the shared content used by that layout. The server then pointcasts each user's layouts to that user, and when the pointcasting is complete, multicasts the shared content on the specified channels. After receiving its pointcast, each Netpage Printer subscribes to the multicast channels specified in the page layouts. During the multicasts, each printer extracts from the multicast streams those objects referenced by its page layouts.

Once a printer has received all the objects to which its page layouts refer, the printer re-creates the fully-populated layout and then rasterizes and prints it.

The server also delivers each page layout to the relevant Netpage Page Server, which may be co-located with the Netpage Publication Server, or may be located elsewhere on the network. Thus the page layouts are persistently archived as Netpages. It is the responsibility of the Netpage Publication Server to preserve the shared objects referenced by the Netpages, to ensure that they are really persistent. It may choose to archive these shared objects elsewhere on the network at any time. The object URIs embedded in the Netpages allow the objects to move.

Under normal circumstances, the printer prints pages faster than they can be delivered. Assuming a quarter of each page is covered with images, the average page has a size of less than 400KB[1]. The printer can therefore hold in excess of 100 such pages in its internal 64MB memory, allowing for temporary buffers etc. The printer prints at a rate of one page per second. This is equivalent to 400KB or about 3Mbit of page data per second, which is similar to the highest expected rate of page data delivery over a broadband network.

Even under abnormal circumstances, such as when the printer runs out of paper, it is likely that the user will be able to replenish the paper supply before the printer's 100-page internal storage capacity is exhausted.

However, if the printer's internal memory does fill up, then the printer will be unable to make use of a multicast when it first occurs. The Netpage Publication Server therefore allows printers to submit requests for re-multicasts. When a critical number of requests is received or a timeout occurs, the server re-multicasts the corresponding shared objects.

Once a document is printed, a Netpage Printer can produce an exact duplicate at any time by retrieving its page layouts from the relevant Netpage Page Server and retrieving the objects to which they refer from the network.

6.7 ON-DEMAND DOCUMENTS

When a Netpage document is requested ad hoc, it is personalized and delivered in much the same way as a periodical. However, since there is no shared content, delivery is made directly to the requesting printer, i.e. without the use of multicast.

1. 267 pixels per inch (ppi) 24-bit RGB, compressed using JPEG 2000 at a ratio of 13:1.

When a non-Netpage document is requested ad hoc, it is not personalized, and it is delivered via a designated Netpage Formatting Server which reformats it as a Netpage document. A Netpage Formatting Server is a special instance of a Netpage Publication Server. The Netpage Formatting Server has knowledge of myriad Internet document formats, including old favorites such as Adobe's Portable Document Format (PDF) [3], and Hypertext Markup Language (HTML) [69]. In the case of HTML, it makes use of the higher resolution of the printed page to present Web pages in a two-column format, with a table of contents and an index of links. By default it automatically includes all Web pages directly linked to the requested page. The user can tune this behavior via a preference.

The Netpage Formatting Server makes standard Netpage behavior, including interactivity and persistence, available on any Internet document, no matter what its origin and format. It hides knowledge of different document formats from both the Netpage Printer and the Netpage Page Server.

7 Security

7.1 CRYPTOGRAPHY

Cryptography is used to protect sensitive information, both in storage and in transit, and to authenticate parties to a transaction. There are two classes of cryptography in widespread use: secret-key cryptography and public-key cryptography. The Netpage Network uses both classes of cryptography.

Secret-key cryptography, also referred to as symmetric cryptography, uses the same key to encrypt and decrypt a message. Two parties wishing to exchange messages must first arrange to securely exchange the secret key.

Public-key cryptography, also referred to as asymmetric cryptography, uses two encryption keys. The two keys are mathematically related in such a way that any message encrypted using one key can only be decrypted using the other key. One of these keys is then published, while the other is kept private. The public key is used to encrypt any message intended for the holder of the private key. Once encrypted using the public key, a message can only be decrypted using the private key. Thus two parties can securely exchange messages without first having to exchange a secret key. To ensure that the private key is secure, it is normal for the holder of the private key to generate the key pair.

Public-key cryptography can be used to create a digital signature. If the holder of the private key creates a known hash of a message and then encrypts the hash using the private key, then anyone can verify that the encrypted hash constitutes the "signature" of the holder of the private key with respect to that particular message, simply by decrypting the encrypted hash using the public key and verifying the hash against the message. If the signature is appended to the message, then the recipient of the message can verify both that the message is genuine and that it has not been altered in transit.

To make public-key cryptography work, there has to be a way to distribute public keys which prevents impersonation. This is normally done using certificates and certificate authorities. A certificate authority is a trusted third party which authenticates the connection between a public key and someone's identity. The certificate authority verifies the person's identity by examining identity documents etc., and then creates and signs a digital certificate containing the person's identity details and public key. Anyone who trusts the certificate authority can use the public key in the certificate with a high degree of certainty that it is genuine. They just have to verify that the certificate has indeed been signed by the certificate authority, whose public key is well-known.

In most transaction environments, public-key cryptography is only used to create digital signatures and to securely exchange secret session keys. Secret-key cryptography is used for all other purposes.

In the following discussion, when reference is made to the *secure* transmission of information between a Netpage Printer and a server, what actually happens is that the printer obtains the server's certificate, authenticates it with reference to the certificate authority, uses the public key-exchange key in the certificate to exchange a secret session key with the server, and then uses the secret session key to encrypt the message data. A *session* key, by definition, can have an arbitrarily short lifetime.

7.2 NETPAGE PRINTER SECURITY

Each Netpage Printer is assigned a pair of unique identifiers at time of manufacture which are stored in read-only memory in the printer and in the Netpage Registration Server database. The first id is public and uniquely identifies the printer on the Netpage Network. The second id is secret and is used when the printer is first registered on the network.

When the printer connects to the Netpage Network for the first time after installation, it creates a signature public/private key pair. It transmits the secret id and the public key securely to the Netpage Registration Server. The server compares the secret id against the printer's secret id recorded in its database, and accepts the registration if the ids match. It then creates and signs a certificate containing the printer's public id and public signature key, and stores the certificate in the registration database.

The Netpage Registration Server acts as a certificate authority for Netpage Printers, since it has access to secret information allowing it to verify printer identity.

When a user subscribes to a publication, a record is created in the Netpage Registration Server database authorizing the publisher to print the publication to the user's default printer. Every document sent to a printer is addressed to a particular user and is signed by the publisher using the publisher's private signature key. The printer verifies, via the registration database, that the publisher is authorized to print the publication on the specified user's default printer and that the printer is the user's default printer. The printer verifies the signature using the publisher's public key, obtained from the publisher's certificate stored in the registration database.

The Netpage Registration Server accepts requests to add printing authorizations to the database, so long as those requests are initiated via a pen registered to the printer.

The user can authorize a Web terminal to print on a printer. This is useful if the user has a Web terminal in the home which is used to locate documents on the Web for printing. The one-time authorization proceeds as follows: the user prints a Web terminal authorization form. The Netpage Registration Server generates a short-lifetime one-time-use id for the Web terminal which is printed on the form, together with the URI of the printer. The Web terminal is used to navigate to a Netpage Registration Server registration site, where the one-time-use id is entered, as well as the URI of the printer. The Web terminal generates a signature public/private key pair. The server allocates a terminal id for the Web terminal, creates and signs a certificate containing the terminal id and the public key, and stores the certificate in the registration database. The URI of the printer, the Web terminal's terminal id, and the private signature key are stored locally on the Web terminal.

Whenever the Web terminal wishes to print on the printer, it sends the printer's designated Netpage Formatting Server a request containing the URI of the document to be printed, together with the terminal id, signed with the Web terminal's private signature key. On receipt of the request and before acting on it, the server verifies, via the registration server, that the terminal is authorized to print on the specified printer, and verifies the signature using the terminal's public key, obtained from the terminal's certificate stored in the registration database.

The user can print a list of current printing authorizations at any time, and revoke any which are being abused.

7.3 NETPAGE PEN SECURITY

Each Netpage Pen is assigned a unique identifier at time of manufacture which is stored in read-only memory in the pen and in the Netpage Registration Server database. The id uniquely identifies the pen on the Netpage Network.

A Netpage Pen can know a number of Netpage Printers, and a printer can know a number of pens. A pen communicates with a printer via a radio frequency signal whenever it is within range of the printer. Once a pen and printer are registered, they regularly exchange session keys. Whenever the pen transmits digital ink to the printer, the digital ink is always encrypted using the appropriate session key. Digital ink is never transmitted in the clear.

A pen stores a session key for every printer is knows, indexed by printer id, and a printer stores a session key for every pen it knows, indexed by pen id. Both have a large but finite storage capacity for session keys, and will forget a session key on a least-recently-used basis if necessary.

When a pen comes within range of a printer, the pen and printer discover whether they know each other. If they don't know each other, then the printer determines whether it is supposed to know the pen, i.e. because the pen belongs to a user who is registered to use the printer. If the printer is meant to know the pen but doesn't, then it initiates the automatic pen registration procedure. If the printer isn't meant to know the pen, then it agrees with the pen to ignore it until the pen is placed in a charging cup, at which time it initiates the registration procedure.

In addition to its public id, the pen contains a secret id and a secret key-exchange key, both intended for one-time use. These are also recorded in the Netpage Registration Server database at time of manufacture. During registration, the printer obtains the secret id from the pen. Because it is transmitted in the clear, it may be intercepted by someone listening in. The printer transmits the id securely to the Netpage Registration Server, which responds securely with the matching key-exchange key, together with a newly-generated secret id and key-exchange key. The printer generates a session key for the pen and transmits it to the pen encrypted using the one-time-use key-exchange key. It also securely transmits the new secret id and key-exchange key to the pen, which saves them for the next registration procedure. They now match the pen's record in the Netpage Registration Server database.

If the secret id transmitted in the clear from the pen to the printer is intercepted and used to retrieve the secret key-exchange key from the Netpage Registration Server before the printer queries the server, then the server rejects the printer's query because the secret id is out-of-date. Thus the printer knows that the pen has been compromised, and recommends that it be returned for repair.

When a previously unregistered pen is first registered, it is of limited use until it is linked to a user. A registered but "un-owned" pen is only allowed to be used to request and fill in Netpage user and pen registration forms [2], i.e. to register a new user to which the new pen is automatically linked, or to add a new pen to an existing user.

The pen uses secret-key rather than public-key encryption because of hardware performance constraints in the pen.

7.4 SECURE DOCUMENTS

The Netpage System supports the delivery of secure documents such as tickets and coupons. The Netpage Printer includes a facility to print watermarks, but will only do so on request from publishers who are suitably authorized. The publisher indicates its authority to print watermarks in its certificate, which the printer is able to authenticate.

The "watermark" printing process uses an alternative dither matrix in specified "watermark" regions of the page. Back-to-back pages contain mirror-image watermark regions which coincide when printed. The dither matrices used in odd and even pages' watermark regions are designed to produce an interference effect when the regions are viewed together - i.e. when looking *through* the printed sheet.

The effect is similar to a watermark in that it is not visible when looking at only one side of the page, and is lost when the page is copied by normal means.

As described in Section 5.8, pages of secure documents cannot be copied using the built-in Netpage copy mechanism. This extends to copying Netpages on Netpage-aware photocopiers.

Secure documents are typically generated as part of e-commerce transactions. They can therefore include the user's photograph which was captured when the user registered biometric information with the Netpage Registration Server, as described in Section 8.

When presented with a secure Netpage document, the recipient can verify its authenticity by requesting its status in the usual way. The unique id of a secure document is only valid for the lifetime of the document, and secure document ids are allocated non-contiguously to prevent their prediction by opportunistic forgers. A secure document verification pen can be developed with built-in feedback on verification failure, to support easy point-of-presentation document verification.

Clearly neither the watermark nor the user's photograph are secure in a cryptographic sense. They simply provide a significant obstacle to casual forgery. Online document verification, particularly using a verification pen, provides an added level of security where it is needed, but is still not entirely immune to forgeries.

7.5 NON-REPUDIATION

In the Netpage System, forms submitted by users are delivered reliably to forms handlers and are persistently archived on Netpage Page Servers. It is therefore impossible for recipients to repudiate delivery.

E-commerce payments made through the system, as described in Section 8, are also impossible for the payee to repudiate.

8 Electronic Commerce Model

8.1 Secure Electronic Transaction (SET)

The Netpage System uses the Secure Electronic Transaction (SET) [42] system as its payment system model. Although SET is not yet widely supported, it is comprehensive and elegant and will probably become dominant in the near future.

SET, having been developed by MasterCard and Visa, is organized around payment cards, and this is reflected in the terminology. However, much of the system is independent of the type of accounts being used.

In SET, cardholders and merchants register with a certificate authority and are issued with certificates containing their public signature keys. The certificate authority verifies a cardholder's registration details with the card issuer as appropriate, and verifies a merchant's registration details with the acquirer as appropriate. Cardholders and merchants store their respective private signature keys securely on their computers. During the payment process, these certificates are used to mutually authenticate a merchant and cardholder, and to authenticate them both to the payment gateway.

SET has not yet been adopted widely, partly because cardholder maintenance of keys and certificates is considered burdensome. Interim solutions which maintain cardholder keys and certificates on a server and give the cardholder access via a password have met with some success [13].

8.2 SET Payments

In the Netpage System the Netpage Registration Server acts as a proxy for the Netpage user (i.e. the cardholder) in SET payment transactions.

The Netpage System uses biometrics to authenticate the user and authorize SET payments. Because the system is pen-based, the biometric used is the user's on-line signature, consisting of time-varying pen position, tilt and pressure. A fingerprint biometric can also be used by designing a fingerprint sensor into the pen, although at a higher cost. The type of biometric used only affects the capture of the biometric, not the authorization aspects of the system.

The first step to being able to make SET payments is to register the user's biometric with the Netpage Registration Server. This is done in a controlled environment, for example a bank, where the biometric can be captured at the same time as the user's identity is verified. The biometric is captured and stored in the registration database, linked to the user's record and to the record of a particular Netpage Pen. The user's photograph is also optionally captured and linked to the record. The SET cardholder registration process is completed, and the resulting private signature key and certificate are stored in the database. The user's payment card information is also stored, giving the Netpage Registration Server enough information to act as the user's proxy in any SET payment transaction.

When the user eventually supplies the biometric to complete a payment, for example by signing a Netpage order form, the printer securely transmits the order information, the pen id and the biometric data to the Netpage Registration Server. The server verifies the biometric with respect to the user identified by the pen id, and from then on acts as the user's proxy in completing the SET payment transaction.

8.3 MICRO-PAYMENTS

The Netpage Network includes a mechanism for micro-payments, to allow the user to be conveniently charged for printing low-cost documents on demand and for copying copyright documents, and possibly also to allow the user to be reimbursed for expenses incurred in printing advertising material. The latter depends on the level of subsidy already provided to the user.

When the user registers for e-commerce, a network account is established which aggregates micro-payments. The user receives a statement on a regular basis, and can settle any outstanding debit balance using the standard payment mechanism.

The network account can be extended to aggregate subscription fees for periodicals, which would also otherwise be presented to the user in the form of individual statements.

8.4 TRANSACTIONS

Whenever a transaction originates through a Netpage form, the form handler has sufficient information, in the shape of the form's unique document instance id, to maintain transaction-specific state information. However, a transaction may also originate through a non-form page such as a printed catalog page, implying, for example, the existence of a virtual "shopping cart". In this case the relevant transaction state information is tied, indirectly, to the unique id of the user.

The Netpage Registration Server maintains an anonymous relationship between a user and a transaction handler via a uniquely numbered transaction, as illustrated in Figure 4. Whenever the user activates a hyperlink tagged with the "transaction" attribute, the Netpage Printer asks the Netpage Registration Server to translate the associated handler id, together with the pen id, into a transaction id. The transaction id is then submitted to the hyperlink transaction handler. For efficiency, the printer caches transaction ids.

The transaction handler maintains state information indexed by transaction id. It is able to retrieve user-specific state information without explicit knowledge of the user.

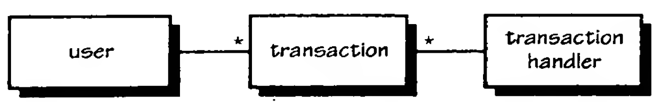


Figure 4. User transaction

APPLICATIONS AND BUSINESS MODELS

9 Applications

The Netpage Network has the potential to subsume a wide variety of applications in both traditional and electronic media. This section sketches the following possibilities:

- personalized subscriptions to newspapers, magazines and comics
- subscriptions to freelance columns and bulletin boards
- on-demand newspapers, magazines and comics
- on-demand flyers and product brochures
- on-demand books
- television infotainment printing
- e-commerce purchasing from online and traditional catalogs
- delivery of statement and invoices, with online payment
- delivery of secure document (tickets, coupons and licenses)
- perfect copying with copyright micro-payments
- mail replacement
- delivery of greeting cards
- form printing, fill-in, and submission
- delivery of e-mail and facsimile
- active business cards
- on-demand document delivery on corporate intranets
- provision of government services
- Web browsing, searching and printing
- photo album creation
- persistent searchable note taking
- exam taking and homework
- computer printing

Many of these applications are described in detail in [2].

9.1 Personalized Subscriptions

The strength of the Netpage Network lies in automatically delivering subscribed periodicals, at a print quality equalling or exceeding that of their traditional counterparts, with editorial content personalized to individual interests, advertising content localized and targeted to individual localities and demographics, and advertising directly linked to detailed product information and product purchasing.

9.1.1 Newspapers and Magazines

The Netpage Network offers a new delivery mechanism to the $24 billion newspaper and newsmagazine market which is both more cost-effective than centralized printing and distribution, and allows more fine-grained targeting of advertising.

The simplest form of news personalization involves selecting a news publication and choosing which daily and weekly sections to receive. The simplest form of advertising personalization is tuned to the demographics associated with the subscriber's ZIP or ZIP+4 code. Even with these entry-level forms of personalization, the Netpage Network offers a compelling distribution model for news.

Users receive paper publications with the usability of their traditional counterparts, but with interactive properties. At the press of an on-page button, a user can print an article giving the background to a news story, print a personalized product brochure, or add a product to the virtual shopping basket.

Any magazine normally printed on lightweight paper stock is equally well-suited to distribution on the Netpage Network. However, since Netpage Printers don't carry heavier paper stocks and don't provide wrap-around binding, the Netpage Network is less well-suited to the distribution of so-called "glossy" magazines.

9.1.2 Freelance Columns and Bulletin Boards

Users can choose to subscribe to individual freelance columns, cartoons, etc. These can be integrated into a user's daily news document, or printed individually. Freelancers can choose to receive micro-payments from their subscribers, freeing them from maintaining their own subscriber databases. The Netpage Network provides mechanisms for handling micro-payments.

Users can also subscribe to the "bulletin boards" of friends: collections of news, announcements, pictures etc., which work much like freelance columns.

9.2 ON-DEMAND PUBLICATIONS

The Netpage Network can deliver, on demand, current and back issues of periodicals normally delivered on subscription, including newspapers, magazines, and comics. To maintain the interactivity of all Netpages ever printed, the Netpage Network keeps all published content online at all times. Unlike the Web, where hyperlinks become unreliable over time, content on the Netpage Network never expires.

9.2.1 Flyers and Product Brochures

The Netpage Network makes high-quality flyers and product brochures instantly available, linked to advertisements and entries in printed catalogs.

Brochures are always up-to-date, and link to e-commerce, e-mail, and automatic telephone call-back. Brochure links can provide "click-through" fees to linking documents, and subsidized printing to users.

Active Netpage advertisements can appear in printed paper publications sold or delivered through traditional outlets, including newspapers and magazines sold on news stands, posters appearing in public places (including in buses, trains and taxis), and books sold in bookshops. Buying guides and travel guide books are particularly suited to containing active Netpage advertising.

9.2.2 Books

Users can obtain the latest best-sellers or rare "out-of-print" (a soon-to-obsoleted term) editions on demand, printed in column format with a text size chosen by the user. A typical 300-page paperback fits on as little as 40 sheets of Letter paper. Slip-on covers are available for robust handling.

Titles which have outlived their copyright period are available for free. Other titles are heavily discounted for Netpage delivery, since publishers avoid the costs of printing, inventory storage, and delivery.

Colorful children's books reproduce immaculately. When they've been loved to death, they can be printed again, and again.

Children's coloring-in books and puzzles are available just when they're needed on a rainy day.

9.2.3 Television Infotainment

Users watching infotainment programs on television can print the associated informational material on their Netpage Printer by pressing the "print" button on their remote control at the appropriate time. The material may be a report on a new medical procedure, the plans for a do-it-yourself bookshelf, or a list of top investment opportunities. It may also be a subscription form for the print publication the program is promoting.

The television and/or remote control is suitably modified or augmented to route the print request to the Netpage Printer. The television or remote control may, for example, include a radio transmitter or transceiver which allows it to communicate with the Netpage Printer. When the print button is pressed, a message is sent to the Netpage Printer which contains the identity of the television channel. The television channel id, date and time together uniquely identify the Netpage document associated with the program at that time. The printer constructs a URN based on the television channel id and determines the address of the television channel's document server via the DNS in the usual way. It then retrieves the document from the document server based on date and time.

A potentially more flexible approach embeds the URN of the document in the television signal itself, for example as a closed caption. In analog television signals there is plenty of room for information in the vertical blanking interval (VBI). The advantage of leveraging support for closed captioning, in particular, is twofold. Closed caption decoders are now standard components in normal-sized television sets, and closed captioning is fully supported in digital television standards that are now being deployed. Other VBI-based schemes require custom decoders and may not survive the transition to digital television.

9.3 E-COMMERCE

9.3.1 Online Purchases

The Netpage Network supports a similar level of online purchasing as the Web, but in paper-based medium which presents like a high-quality printed catalog.

A user can navigate the retailer's online Netpage catalog, printing catalog pages as they're needed and adding items to a virtual shopping trolley. The contents of the shopping cart can be listed at any time, and items can be struck from the list at the stroke of the pen. Pressing a "proceed to checkout" button at any time elicits a completed order form just waiting for the user's signature. The payment card account number is securely shown in the usual 1234 56** **** *789 format. The user's signature authorizes the payment.

9.3.2 Catalog Purchases

Rather than buying from an online Netpage catalog, the user can select items from a traditionally-printed catalog which contains active Netpage links.

9.3.3 Statements and Invoices

Statements and invoices can be securely and auditably delivered, and can be automatically filled in with the user's default payment details without the sender knowing those details.

The user's signature can authorize the payment as normal.

9.3.4 Secure Documents

Retailers can securely issue tickets and coupons over the Netpage Network, printed with difficult-to-forge watermarks.

Agencies of various kinds can issue licenses printed with watermarks and the user's own photograph.

As described earlier, recipients can verify the authenticity of secure documents using a standard Netpage Printer or a special verification pen.

9.3.5 Copyright Copying

Any printed version of a Netpage document becomes an easy means to printing another perfect copy. When a copy is made, the Netpage Network can automatically transfer a micro-payment from the copier to the copyright holder.

Trivial copyright fees are universally respected but seldom paid because of the inconvenience. The Netpage Network offers micro-payment convenience and the quality of an original copy.

9.4 COMMUNICATION

9.4.1 Mail

The Netpage Network, once widely subscribed, can be used to deliver numerous instances of regular mail-outs, particularly statements and invoices as discussed in Section 9.3.3. As already described, the Netpage System only delivers mail from sources authorized by the recipient.

The United States Postal Service delivers 107,000,000,000 pieces of First Class mail each year [64], a large number of which are regular in nature.

9.4.2 Greeting Cards

A user can select a greeting card from an online catalog, add a handwritten message, and dispatch it via the Netpage Network. Cards can be addressed to other Netpage users, and to normal postal addresses. In the latter case the card is printed at the service center closest to the recipient, automatically placed in an envelope, and mailed through the local mail system.

Netpage users can choose to receive cards from anyone, or only from authorized friends.

9.4.3 Forms

Forms of all kinds can be printed on the Netpage Printer, filled in by hand, and submitted directly over the Netpage Network. Submission is secure and cannot be repudiated.

Handwriting is automatically recognized by the system. The digital ink of the handwriting is attached to the form in case a human clerk needs to re-interpret the handwriting. Automatic "handwriting bots" on the network can assist with the recognition task, automatically giving the user semi-intelligent feedback to elicit disambiguation.

Any interactive Netpage "application", including e-commerce and e-mail, uses forms of various kinds.

9.4.4 E-Mail

E-mail forms can be printed on demand and filled in by hand. The handwritten name of the recipient is converted to facilitate delivery, but the rest of the message is delivered as digital ink, just as the user intended. If the recipient is computer-based rather than Netpage-based, all of the handwriting can be automatically converted, with the digital ink sent as an attachment (since it may contain hand-drawn diagrams etc).

Netpage users can choose to receive e-mail from anyone, or only from authorized friends.

Each user maintains a list of contacts which allows e-mail to be addressed by name. Users never deal with Netpage user ids directly. If the specified name is ambiguous, the system prints a list of choices. The use of nicknames helps to avoid ambiguity.

9.4.5 Business Cards

Each user's business card can act as a convenient one-time e-mail authorization token. It is encoded as a Netpage and contains a button which, when pressed by the card recipient, adds the card owner to the recipient's contact list and authorizes the recipient to send the card owner e-mail.

The business card can also contain buttons which link to company information, personal information, and even product information if the user is a salesperson.

The Netpage Network provides a bureau service which allows users to order active Netpage business cards.

9.4.6 Facsimile

Facsimile forms can be printed on demand and filled in by hand. The handwritten telephone number is converted to facilitate delivery, but the rest of the message is delivered as bitmapped digital ink, just as the user intended.

9.4.7 Conferencing

Geographically distributed participants can communicate graphical ideas via a shared Netpage "whiteboard" while engaged in a telephone conference. Every time a participant adds a modification or a new diagram to the whiteboard the other participants are given a new copy of the page. New pages can be started at any time, but older pages or older versions of pages can still be marked up and re-distributed. Each participant ends up with a full history of the interaction. Multiple colors and line styles can be chosen from a palette.

9.5 CORPORATE AND GOVERNMENT

9.5.1 Corporate Intranets

An organization can use a private intranet-based Netpage Network to implement a document repository and efficiently distribute documents on demand.

9.5.2 Government Services

Government can provide access to services via the Netpage Network. The network can obviate the need to visit government offices to obtain forms and submit forms, and the network can be used to efficiently deliver the results of submissions.

9.6 PERSONAL

9.6.1 Web Browsing, Searching and Printing

Users can browse the World Wide Web via their Netpage Printer using paper and pen as the user interface. Netpage forms can provide emulation of HTML forms. Only dynamic media objects may fail to print meaningfully.

A Netpage Printer can be the ideal output device for documents encountered while browsing the Web, whether the browsing is terminal-based or Netpage-based. An increasing number of print-ready documents are being published on the Web.

Entire Web sites can be compactly formatted for print, since a printed page has a much greater information-bearing capacity than a computer screen, and perused in a more leisurely fashion. The Netpage Formatting Server automatically adds a table of contents and an index of referenced pages.

Links on Web pages remain active when printed on Netpages. Clicking on a link causes the linked Web page to be printed.

Web pages and search results are printed on both sides of the paper, allowing one sheet of paper to hold the equivalent of ten SVGA screens of information. Thus the user sees 200 links on paper, compared with only 20 on a computer screen.

9.6.2 Photo Album

A Netpage Printer can be enhanced with an infrared data connection (IrDA) to allow it to accept images wirelessly from a digital camera. Alternatively, digital cameras with USB ports can be directly connected. Images can be automatically archived on the Netpage Network, and individual photos, both regular and poster-sized, can be printed at photo quality on the printer.

Smart layout software can assist with the interactive creation of photo album pages, ready for insertion in a cumulative family album.

9.6.3 Notebook

A pre-printed Netpage notebook can be used to capture handwritten notes in a persistent fashion. Notes are captured on the network and are optionally timestamped and signed by a certificate authority to provide undeniable proof of priority, for example with respect to a

patent application. Since the handwriting is recognized and converted to text for indexing purposes, the notes can also be searched by keyword, date, etc.

9.6.4 Examination Papers and Homework

Students can sit exams in a controlled manner while physically distant from the classroom.

An examination paper can be delivered at a specific time to a distributed collection of students. The students can then be required to answer the questions on the paper within the allotted time. Their answers can be automatically captured at the end of the allotted time using the "blackboard" form model, and in many cases automatically graded. Once again, unrecognized input can be routed to a human operator.

Multiple choice questions can be automatically graded.

Homework in general can be distributed and captured in a similar way.

9.6.5 Computer Printing

A Netpage Printer can be the ideal output device for a personal computer or workstation, whether directly-connected or on a local-area network. The speed of the Netpage Printer and the quality of its output make it attractive to existing computer users.

The Netpage Printers have a USB port as a standard feature.

10 Business Models

10.1 System Principles

The Netpage Network leverages the open technology and extensive infrastructure of the Internet. The widespread acceptance and growth of the Netpage Network is predicated on open competition rather than monopolistic practices.

However, to provide an incentive to early investors, semi-exclusive licenses to Memjet-based Netpage Printer designs will be offered, as well as licenses to manufacture paper and ink consumables.

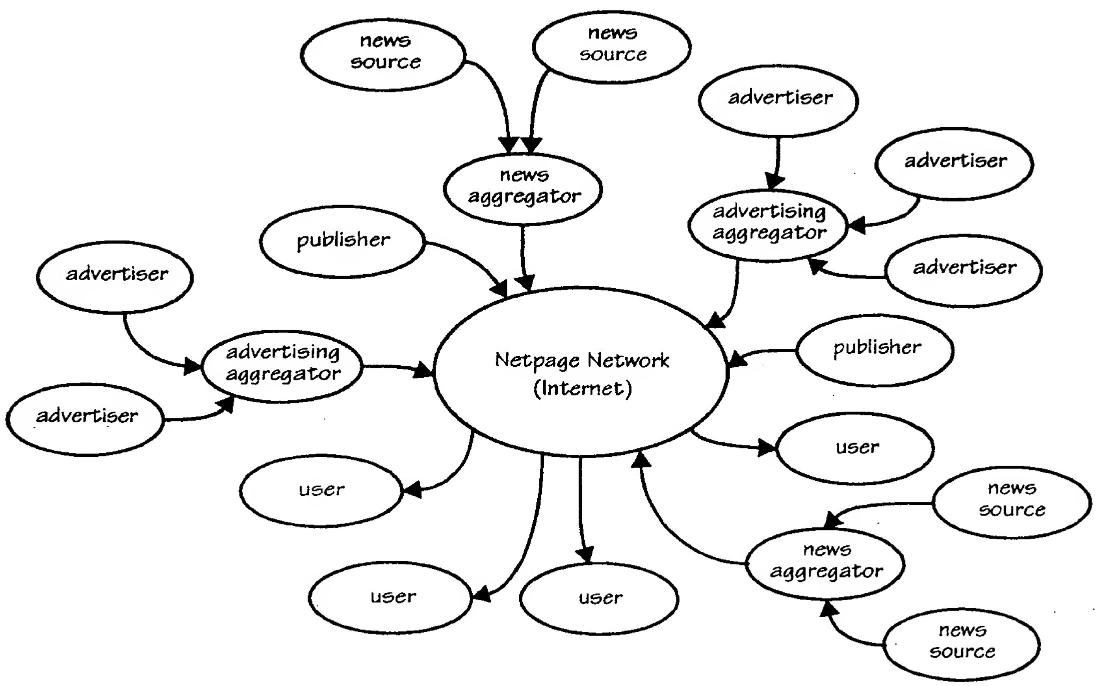


Figure 5. Open structure of Internet-based Netpage Network

The network supports any number of independent participants, some of which have complementary roles, and some of which compete. The open structure of the network is illustrated in Figure 5. Content-related participants include the following:

- news sources
- publishers
- news aggregators
- freelance artists, writers, cartoonists
- direct mailers
- advertisers
- advertising aggregators
- banks
- merchants

Infrastructure-related participants include the following:

- server suppliers
- network storage providers
- communications carriers
- Internet service providers (ISPs)
- printer dealers
- printer installation and servicing companies
- ink and paper consumables dealers
- consumables delivery companies

Technology-related participants include the following:

- research and development
- chip makers (printheads, controllers, QA)
- printer manufacturers
- ink and paper consumables manufacturers

The strength of the network lies in the fact that publication and delivery are completely decoupled. This allows the delivery infrastructure to grow independently of the participation of publishers.

10.2 BOOTSTRAPPING

Because consumers are unlikely to be motivated to acquire a Netpage Printer until a variety of publications and services are available, and because publishers will wait for an installed base before participating, the key to bootstrapping the network is to bundle the printer with a publication or service subscription, and possibly trimming profit margins in the growth stage.

There are several ways the manufacturing cost of the Netpage Printer, assumed to be well below $100, can be subsidized. Printer-based distribution can eliminate existing distribution costs, offsetting the printer cost. The printer can provide a new mechanism for delivering advertising, with advertising profits offsetting the printer cost. And the printer cost can be built into the subscription fee for a publication or service.

The cost of printing and delivering a newspaper normally exceeds the price of subscription [26,50]. The real profit lies in the advertising. The cost of the Netpage Printer is easily exceeded by one years' cost savings, allowing a Netpage subscription, including a "free" printer, to be priced lower than a traditional subscription. A Netpage subscription offered to a customer already on the network would be priced correspondingly lower still.

If the publication or service delivered via the Netpage Printer is sufficiently lucrative, then the publisher or provider may be able to subsidize not only the printer itself, but also its running costs. This can include Internet access, paper and ink consumables, and servicing. Demographics-informed advertising may fall into this category. The more information customers reveal about themselves, the greater the value of the advertising to the advertisers, and so the greater the level of subsidization that can take place.

Early investors who subsidize the installation of Netpage Printers may be able to recover the investment and turn a profit merely by charging other publishers an access fee to the

printers they "own", perhaps for an interim period after installation, according to a network-wide agreement. They may also be able to earn commissions on click-throughs and e-commerce transactions originating on pages printed on "their" printers.

Similar approaches are already emerging in the general Internet market. In the "FreePC" and related models [36], personal computers are bundled with Internet access, and the whole package is fully or partially subsidized by advertising and e-commerce.

Most content-related participants in the Netpage Network, and even Internet service providers, can benefit from directly investing in Netpage Printer deployment.

10.3 MATURITY

Many of the bundling approaches are likely to remain applicable once the network becomes widely subscribed. It is possible that the bundling of the appliance (i.e. the Netpage Printer) with the service (be it Internet access or a publication subscription) will remain the dominant means of distributing the appliance, as it is in the cellular telephone market, and as an increasing number of companies, including IBM [17], are beginning to believe it should be in the personal computer market.

10.3.1 News Publishers and News Aggregators

News publishers with strong brands are likely to be able leverage those brands on the Netpage Network. They have an incentive to do so quickly to prevent newcomers from filling the vacuum and capturing the attendant advertising revenue.

News publishers also have an incentive to migrate to the Netpage Network because it allows them to offer the more fine-grained targeting that advertisers are increasingly demanding, and which they are increasingly seeking elsewhere.

News publishers who create content rather than simply aggregating other sources have a significant advantage, since they offer both unique content and an editorial voice. Users are more likely to choose a single news publication whose content and editorial orientation they find useful, than specifying to a news aggregator how to glue together a number of disparate news sources. And any sufficiently strong news publication brand is unlikely to make its content available to an aggregator, since the aggregator will be taking a proportion of advertising and e-commerce revenue.

The Netpage Network, like the Web, offers lower barriers to entry than traditional publishing media, and this naturally stimulates greater diversity. However, the geographic independence of the network, coupled with built-in mechanisms for localization of publications, allows international, national and regional news publications to more easily compete in local news markets.

The strength of a traditional local news publication lies partly in its local news content and partly in its local retail advertising and classified advertising content. Aggregation of classified advertising is already happening on the Web, and the Netpage Network will make the same thing possible for local retail advertising. Local news publications are therefore likely to be excluded from the direct capture of local advertising, and may instead transform themselves into news gatherers feeding localized editions of larger publications.

10.3.2 Advertising Aggregators

The Netpage Network promises to be the most effective advertising medium ever conceived. It combines the editorial and print quality of traditional publications with arbitrarily finely targeted advertising, and provides a direct link between advertising, product information, and purchasing.

Because personalization and localization are handled automatically by Netpage Publication Servers, an advertising aggregator can provide arbitrarily broad coverage of both geography and demographics. The subsequent disaggregation is efficient because it is automatic.

This makes it more cost-effective for publishers to deal with advertising aggregators than to directly capture advertising. Even though the advertising aggregator is taking a proportion of advertising revenue, publishers may find the change profit-neutral because of the greater efficiency of aggregation.

Because of the finer targeting supported by the Netpage Network, publishers and advertising aggregators have a larger advertising space to sell, leading to greater profits. The linking between advertising, detailed product information, and purchasing, and the corresponding measurability of consumer behavior, leads to greater profits from click-through fees and e-commerce commissions, benefiting publishers and advertising aggregators alike.

Added revenue from these fees and commissions may even allow users' costs - printer, ink, paper, and Internet access - to be fully subsidized.

REFERENCES

11 Silverbrook References

[1] Silverbrook Research, *Memjet*, 1998
[2] Silverbrook Research, *Netpage Applications*, v0.1, 1999

12 Other References

[3] Adobe Systems Inc., *Portable Document Format Reference Manual*, version 1.2, November 27, 1996

[4] Bank of Montreal Economics Department, "Newsprint Market Update", Forest Products, May 1998

[5] Bender, W., "Read all about it in the Daily You", Communicating Business, Forward Publishing, London, Winter 1994/95 (http://nif.www.media.mit.edu/papers/forward.html)

[6] Brand, S., "Personal Newspaper", in *The Media Lab*, Penguin Books, 1988, pp.36-39

[7] Bray, T., "News Wire Services Heading for XML", 12 August 1998, http://xml.com/xml/pub/98/08/nitf.html

[8] Cable Datacom News, "Cable Modem Customer Count To Top 500,000 At Year's End", December 1998, http://www.cabledatacomnews.com/dec98-1.htm

[9] Cable Foreman Xtra, "WDM boosts fiber-optic capacity", May 1998, http://www.broadband-guide.com/cbl4man/eng/eng5981.html

[10] Chesnais, P.R., M.J. Mucklo and J.A. Sheena, "The Fishwrap Personalized News System", *IEEE Second International Workshop on Community Networking*, Integrating Multimedia Services to the Home, 1995

[11] Cisco Systems, "Cisco 605 Personal PCI ADSL Modem", http://www.cisco.com/warp/public/728/605

[12] Cisco Systems, "Cisco 6100 Series advanced DSL Access Multiplexer", http://www.cisco.com/warp/public/728/6100

[13] Clark, T., "Visa, Mastercard try to revive SET", *CNET NEWS.COM*, 12 May 1999, http://www.news.com/News/Item/0,4,0-36435,00.html@st.ne.180.head

[14] Click-Through, http://www.clickthrough.com

[15] CNN Interactive, "With 'electronic ink', a printed page can be a work in progress", http://cnn.com/TECH/science/9812/23/electronic.ink.ap/index.html

[16] Cook, P.S., D. Gomery, and L.W. Lichty (eds.), *The Future of News - Television, Newspapers, Wire Services, Newsmagazines*, The Woodrow Wilson Center Press, 1992

[17] Crothers, B., "IBM eyes new way to sell PCs", *CNET NEWS.COM*, 14 May 1999, http://www.news.com/News/Item/0,4,0-36508,00.html@st.ne.ni.lh

[18] Daniel, R., and Mealling, M., "Resolution of Uniform Resource Identifiers using the Domain Name System", *Internet Engineering Task Force RFC2168*, June 1997

[19] Doane, W.J., "A Comparison of Display Technologies for E-Books", http://www.jmc.kent.edu/futureprint/1998fall/doane.htm

[20] E Ink, http://www.eink.com

[21] Ericsson, *3G Cellular Technology*, http://www.ericsson.se/wireless/products/modsys/3rdgen/subpages/3gcell

[22] E&P Interactive, "Online News Reading Soars", *E&P Interactive*, 12 June 1998, http://www.mediainfo.com/ephome/news/newshtm/stories/061298n4.htm

[23] E&P Interactive, "Audience Rather than Circulation", *E&P Interactive*, 24 July 1998, http://www.mediainfo.com/ephome/news/newshtm/stories/072498n4.htm

[24] E&P Interactive, "Newspapers Near Death, Gates Says", *E&P Interactive*, 5 February 1999, http://www.mediainfo.com/ephome/news/newshtm/stories/020599n1.htm

[25] Fitzgerald, M., "Newspapers Re-think Microzoning", *E&P Interactive*, 14 January 1999, http://www.mediainfo.com/ephome/news/newshtm/stories/011499n3.htm

[26] Gannet Co. Inc., *Consolidated Statements of Income 1998*, http://www.gannet.com/annual/ar98/financials/income.htm

[27] Handley, K., "All the World's a Page", http://www.ingersoll-rand.com/compair/oct_nov_97/octnovco/pag_1.htm

[28] Hughes Network Systems, "Hughes Spaceway - Wireless Broadband on Demand", http://www.hns.com/spaceway/spaceway.htm

[29] INRIA, "Light-weight Reliable Multicast Protocol", http://monet.inria.fr/lrmp/lrmp.html

[30] INRIA, "WebCanal White Paper - Global Information Broadcast", http://monet.inria.fr/white/index.html

[31] IP Multicast Initiative (IPMI), *Higher Level Protocols used with IP Multicast*, 1997, `http://www.ipmulticast.com/...`
[32] IPTC - NAA, *News Industry Text Format*, version 2.0b1, June 1998
[33] ISO/IEC JTC1/SC29/WG1 (Joint Photographic Experts Group), *Call for contributions for JPEG 2000: Image Coding System*, `http://www.jpeg.org/public/wg1n505.pdf`
[34] ISO/IEC JTC1/SC29/WG1 (Joint Photographic Experts Group), *JPEG 2000 requirements and profiles version 4.0*, `http://www.jpeg.org/public/wg1n1105.pdf`
[35] Kent Displays, `http://www.kentdisplays.com`
[36] Krause, J.K., "Two More 'Free' PC Firms Pin Hopes on E-commerce", *The Industry Standard*, 1 April 1999, `http://wwwthestandard.net/articles/article_print/0,1454,4073,00.html`
[37] Lambert, P., "Multicast Debuts on Cable Modems", *ZDNN*, 5 August 1997, `http://www5.zdnet.com/zdnn/content/inwk/0426/inwk0006.html`
[38] Liebeskind, K., "Newspapers Continue to Report Higher Revenues", E&P Interactive, 8 January 1998, `http://www.mediainfo.com/ephome/news/newshtm/stories/010899n1.htm`
[39] Loberg, C.J., "Testing the Future of All-Optical Networks", Lightwave Xtra, March 1998, `http://www.broadband-guide.com/lw/reports/report3981.html`
[40] McCanne, S., "Overview of Reliable Multicast Protocols", *Matters Mbone Workshop*, SIGCOMM '96, August 1996, `http://gaia.cs.umass.edu/sigcomm_mcast/talk1.html`
[41] McCanne, S., "SRM: Work in Progress", *Matters Mbone Workshop*, SIGCOMM '96, August 1996, `http://gaia.cs.umass.edu/sigcomm_mcast/talk2.html`
[42] MasterCard/Visa, *SET Secure Electronic Transaction Specification, Book 1: Business Description*, Version 1.0, 31 May 1997, `http://www.setco.org/set_specifications.html`
[43] Medin, M., "Data Over Cable and the @Home Network", `http://www.academ.com/nanog/feb1996/data.over.cable.html`
[44] MIT Media Lab, *News in the future*, `http://nif.www.media.mit.edu/nif.html`
[45] MIT Media Lab, *Fishwrap Documentation Center*, `http://fishwrap-docs.www.media.mit.edu/docs`
[46] MIT Media Lab, *Micromedia - Electronic Paper* `http://www.media.mit.edu/micromedia/elecpaper.html`
[47] My Yahoo, `http://my.yahoo.com`
[48] Negroponte, N., "Less is More", in *Being digital*, Vintage Books, 1996, pp.149-159
[49] New Media Communication, *CyberStream Fast Internet & Data Broadcasting System via Satellite*, `http://www.nmc.com`
[50] New York Times Co., *Consolidated Statements of Income 1998*, `http://www.nytco.com/pdf/consol1_1998.pdf`
[51] Neuwirth, R., "Inserts Outpace ROP", *E&P Interactive*, 24 September 1998, `http://www.mediainfo.com/ephome/news/newshtm/stories/092598n1.htm`
[52] Newspaper Association of America (NAA), *Facts About Newspapers 1998* `http://www.naa.org/info/facts`
[53] Noack, D., "Online News Audience Becomes More Mainstream", *E&P Interactive*, 15 January 1999, `http://www.mediainfo.com/ephome/news/newshtm/stories/011599n2.htm`
[54] Schwartz, E.I., "Advertising Webonomics 101", *Wired* magazine, February 1996
[55] Shardanand, U. and Maes, P., "Social Information Filtering: Algorithms for Automating 'Word of Mouth'", *Proceedings of ACM Computer-Human Interaction (CHI) 1995*
[56] Steinberg, D., "Broading the Bandwidth to the Home in USA and Canada", 15 May 1997, `http://www.specialty.com/hiband/mi-int.html`
[57] Sterne, J., *Advertising on the Web*, Que 1997
[58] Stevens, W.R., *TCP/IP Illustrated, Volume 1 - The Protocols*, Addison-Wesley 1994
[59] Stone, M., "Newspapers Must Change, Fast - API Cites Serious Threat of Online Classifieds", *E&P Interactive*, 24 July 1998, `http://www.mediainfo.com/ephome/news/newshtm/stories/072498n1.htm`
[60] Sun Microsystems, `http://www.sun.com`
[61] Tappert, C.C., Suen, C.Y., Wakahara, T., "The State of the Art in On-Line Handwriting Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.12, No.8, August 1990

[62] Teledesic, *Teledesic Overview*, http://www.teledesic.com/overview.html
[63] The New York Times on the Web, http://www.nytimes.com
[64] United States Postal Service, *Postal Facts May 1998*, http://www.usps.gov/history/pfact98.htm
[65] Wallace, G.K., "The JPEG Still Picture Compression Standard", Communications of the ACM, Vol.34, No.4, April 1991, pp.30-44
[66] Weinberger, D., "The Daily Me? No, the Daily Us", *Wired* magazine, April 1995
[67] World Wide Web Consortium, *Extensible Markup Language (XML) 1.0*, W3C Recommendation, 10 February 1998, http://www.w3.org/TR/1998/REC-xml-19980210
[68] World Wide Web Consortium, *Extensible Stylesheet Language (XSL) Specification*, W3C Working Draft, 21 April 1999, http://www.w3.org/TR/WD-XS/
[69] World Wide Web Consortium, *HTML 4.0 Specification*, W3C Recommendation, 24 April 1998, http://www.w3.org/TR/REL_html40/
[70] World Wide Web Consortium, *Scalable Vector Graphics (SVG) Specification*, W3C Working Draft, 12 April 1999, http://www.w3.org/TR/WD-SVG/
[71] Zollman, P.M., "1999 Could make or Break News Web Sites", *E&P Interactive*, 16 December 1998, http://www.mediainfo.com/ephome/news/newshtm/stories/121698n1.htm

Appendix 2
ages 70)

Netpage
Applications

draft version 0.1, 7 September 1999

Silverbrook Research Pty Ltd
393 Darling Street, Balmain
NSW 2041 Australia
Phone: +61 2 9818 6633
Fax: +61 2 9818 6711
Email: info@silverbrook.com.au

Confidential

Document History

0.1d	7 September 1999	Paul Lapstun Kia Silverbrook Ann Morrison	Initial draft issue.

Contents

APPLICATION ARCHITECTURES

1 Netpage Applications

The Netpage System can be used to deliver many of the same interactive computer applications normally delivered via networked computer terminals such as Web terminals.

In the Web paradigm, a series of Web "pages" displayed on the terminal screen provide the user interface to the application. In the Netpage paradigm, a series of printed Netpages provide the user interface to the application.

The following sections define a set of core Netpage applications, including user and pen registration, news subscription, electronic mail (e-mail), and various examples of electronic commerce (e-commerce). In each case the application is described from the point of view of the user, in terms of the Netpage user interface seen by the user and the object model implicitly manipulated by the user through the user interface.

1.1 Object Model Description

Each object model is described using a Unified Modeling Language (UML) class diagram [1,2]. A class diagram consists of a set of object classes connected by relationships. Two kinds of relationships are of interest here: associations and generalizations. An association represents some kind of relationship between objects, i.e. between instances of classes. A generalization relates actual classes, and can be understood in the following way: if a class is thought of as the set of all objects of that class, and class A is a generalization of class B, then B is simply a subset of A. The UML does not directly support second-order modelling - i.e. classes of classes.

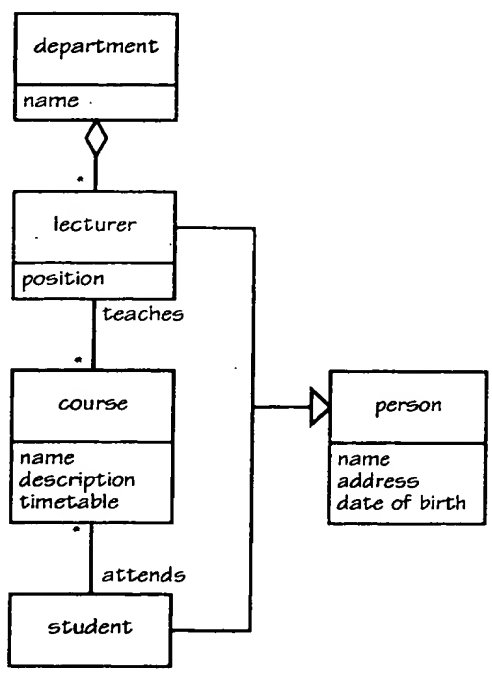


Figure 1. Illustrative UML class diagram

Each class is drawn as a rectangle labelled with the name of the class. It contains a list of the attributes of the class, separated from the name by a horizontal line, and a list of the

operations of the class, separated from the attribute list by a horizontal line. In the class diagrams which follow, however, operations are never modelled.

An association is drawn as a line joining two classes, optionally labelled at either end with the multiplicity of the association. The default multiplicity is one. An asterisk (*) indicates a multiplicity of "many", i.e. zero or more. Each association is optionally labelled with its name, and is also optionally labelled at either end with the role of the corresponding class. An open diamond indicates an aggregation association ("is-part-of"), and is drawn at the aggregator end of the association line.

A generalization relationship ("is-a") is drawn as a solid line joining two classes, with an arrow (in the form of an open triangle) at the generalization end.

When a class diagram is broken up into multiple diagrams, any class which is duplicated is shown with a dashed outline in all but the main diagram which defines it. It is shown with attributes only where it is defined.

Figure 1 illustrates the subset of the UML class diagram notation used here.

1.2 USER INTERFACE DESCRIPTION

Each application user interface flow is illustrated as a collection of documents linked by command arrows. A command arrow indicates that the target document is printed as a result of the user pressing the corresponding command button on the source page. Some command arrows are labelled with multiple commands separated by slashes ('/'s), indicating that any one of the specified commands causes the target document to be printed. Although multiple commands may label the same command arrow, they typically have different side-effects.

In application terms, it is important to distinguish between Netpage documents and Netpage forms. Documents contain printed information, as well as command buttons which can be pressed by the user to request further information or some other action. Forms, in addition to behaving like normal documents, also contain input fields which can be filled in by the user. They provide the system with a data input mechanism.

It is also useful to distinguish between documents which contain generic information and documents which contain information specific to a particular interaction between the user and an application.

Each document type has a distinct iconic representation as shown in Figure 2.

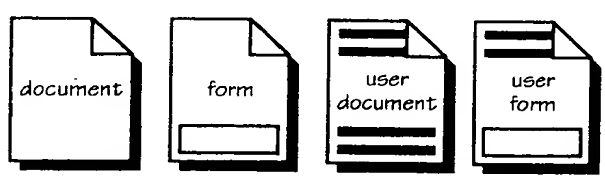


Figure 2. Document icons used in user interface flows

Generic documents may be pre-printed publications such as magazines sold at news stands or advertising posters encountered in public places. Forms may also be pre-printed, including, for example, subscription forms encountered in pre-printed publications. They

may, of course, also be generated on-the-fly by a Netpage Printer in response to user requests.

User-specific documents and forms are normally generated on the fly by a Netpage Printer in response to user requests.

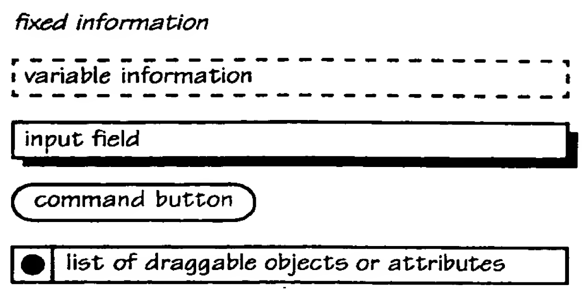


Figure 3. Page layout elements

The Netpages which participate in a user interface flow are further described as abstract page layouts. A page layout may contain fixed information, variable information, input fields, command buttons, and lists of draggable objects and attributes. Each type of layout element has a unique style to differentiate it from the others, as illustrated in Figure 3.

When a user interface flow is broken up into multiple diagrams, any document which is duplicated is shown with dashed outlines in all but the main diagram which defines it.

1.3 NETPAGE USER INTERFACE DESIGN GUIDELINES

Like a Web application, a Netpage application is forms-oriented and transaction-driven. The application is largely decoupled from the user interface in the sense that it is only notified when the user submits a form, not while the user is filling in the form. When the time it takes to deliver a form is short, a transaction-driven system can approximate the behavior of a truly interactive system. Such is often the case with the Web. With the Netpage System, however, the time it takes to deliver a form is longer, but more importantly, a form is not delivered *in situ*, but in the shape of another physical piece of paper. The new page must be physically collected from the printer, which may not be proximate to the user, and even if the actual printing cost of the page is small, the user may perceive the user interface as being wasteful.

It therefore behooves a good Netpage application user interface to be more sparing in the printing of intermediate pages than an equivalent screen-based application, and to ensure that the content of each printed page is valuable to the user whenever a page is printed.

1.3.1 Input Feedback

As with any computer application, it is important to provide the user with immediate feedback on all input, so that the evolving state of the transaction is visible and not hidden.

When a user fills in a Netpage form, immediate feedback is provided because the user uses the marking nib of the Netpage Pen. This applies to input in textboxes, checkboxes, signature fields and the like. When the user presses a command button, there is no immediate feedback on the page since the user uses the non-marking nib of the pen, but feedback is

nonetheless provided in the shape of the next form in the user interface series, or a transaction receipt, or some other requested document.

The user also uses the marking nib of the pen to input commands which don't require immediate interpretation, such as text editing markups.

The following are examples of input typically made with the marking nib:

- handwritten text
- drawing
- checkbox mark
- signature
- text editing command (strikeout etc.)
- drag-and-drop (from list of objects, etc.)
- drag-and-lasso (from palette of colors, line styles, etc.)

The following are examples of input typically made with the non-marking nib:

- selection by circumscription
- button press

1.3.2 A Minimum Unit of User Interface Output

A Netpage Printer prints at a constant page rate, independent of whether pages are printed on both sides or not, and independent of the pages' actual content.

A single page printed on both sides at high resolution in full color is therefore a useful minimum unit of user interface output. This minimum unit of output has an information-bearing capacity equivalent to ten SVGA computer screens. So although a Netpage application is designed to produce less frequent output than the equivalent screen-based application, it has ten times the "display" area of the screen-based application when it does produce output.

Furthermore, previous pages printed as part of the user interface flow continue to act as part of the user interface, and multiple applications can be active simultaneously without competing for screen space and forcing the user to switch between them on the screen.

2 Directory Navigation

A directory is an organized list of named objects. It may be presented in many ways: sorted alphabetically, arranged into named groups or topics, etc. The default presentation is specific to each kind of directory. A directory may also be searched.

For the purposes of navigation, an n-ary index tree is built above the directory, with each node containing enough index entries to fill two printed pages (i.e. a double-sided sheet). If two pages hold k entries, and the entire directory contains m entries, then the number of levels in the index is $log_k m$. For example, if two pages hold 200 entries, and the entire directory contains 1,000,000,000 entries, then the number of levels in the index is 4.

Each index entry specifies the starting and ending names on the two pages it is linked to, separated by "to", e.g. "**Aardvark to Axolotl**", "**Ayatollah to Bernoulli**", etc. Each index entry acts as a hyperlink to the corresponding lower-level index node. The layout of the generic directory index page is shown in Figure 7.

If the entire directory fits on two pages, then it is presented directly without an intervening index tree.

The indexed directory class diagram is shown in Figure 20.

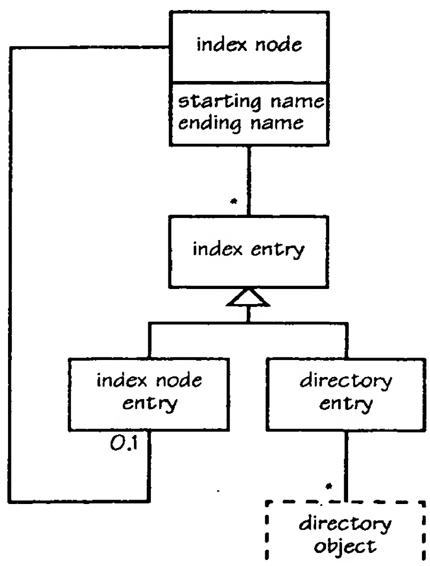


Figure 4. Indexed directory class diagram

The presentation of the index entries of a terminal (bottom-level) index node, i.e. a node which refers to actual named objects, is specific to each kind of directory.

Each page contains a <top> button which prints the root node of the index, an <up> button which prints the page's parent node in the index, and <first>, <previous>, <next> and <last> buttons which print the first, previous, next and last nodes, respectively, of the page's index level. Particular navigation buttons don't appear if they have no meaning for the current page, e.g. the page for the first index node in an index level doesn't have <first> and <previous> buttons.

Each page also contains a <search> button which generates a search form which allows the directory to be searched. The contents of the search form are specific to each kind of directory.

The directory navigation user interface flow is illustrated in Figure 5.

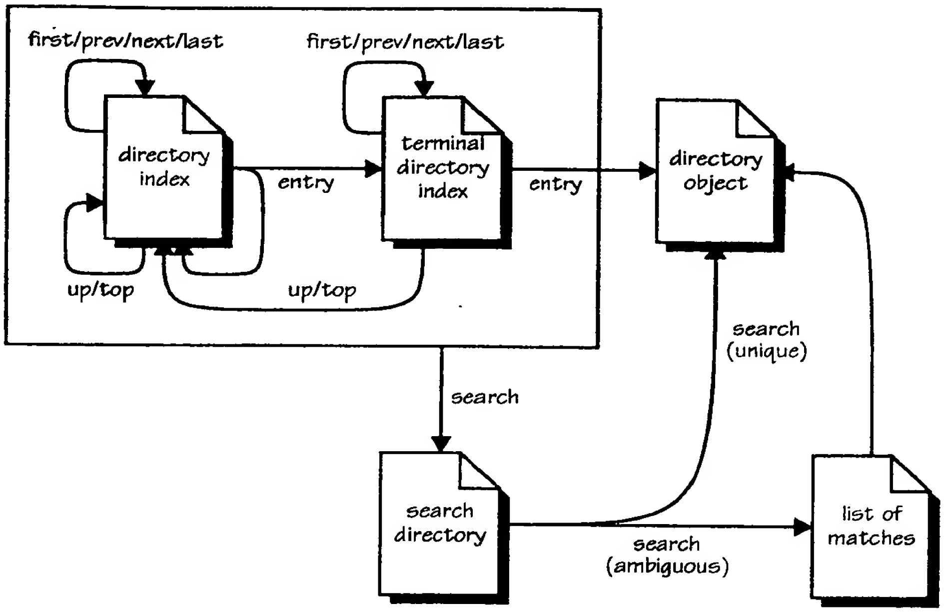


Figure 5. Directory navigation user interface flow

The meaning of the standard navigation buttons on the directory index page is described in Figure 6.

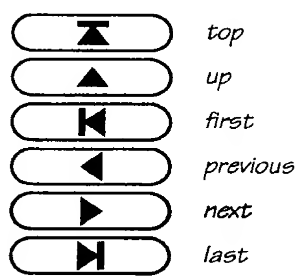


Figure 6. Standard directory index navigation button

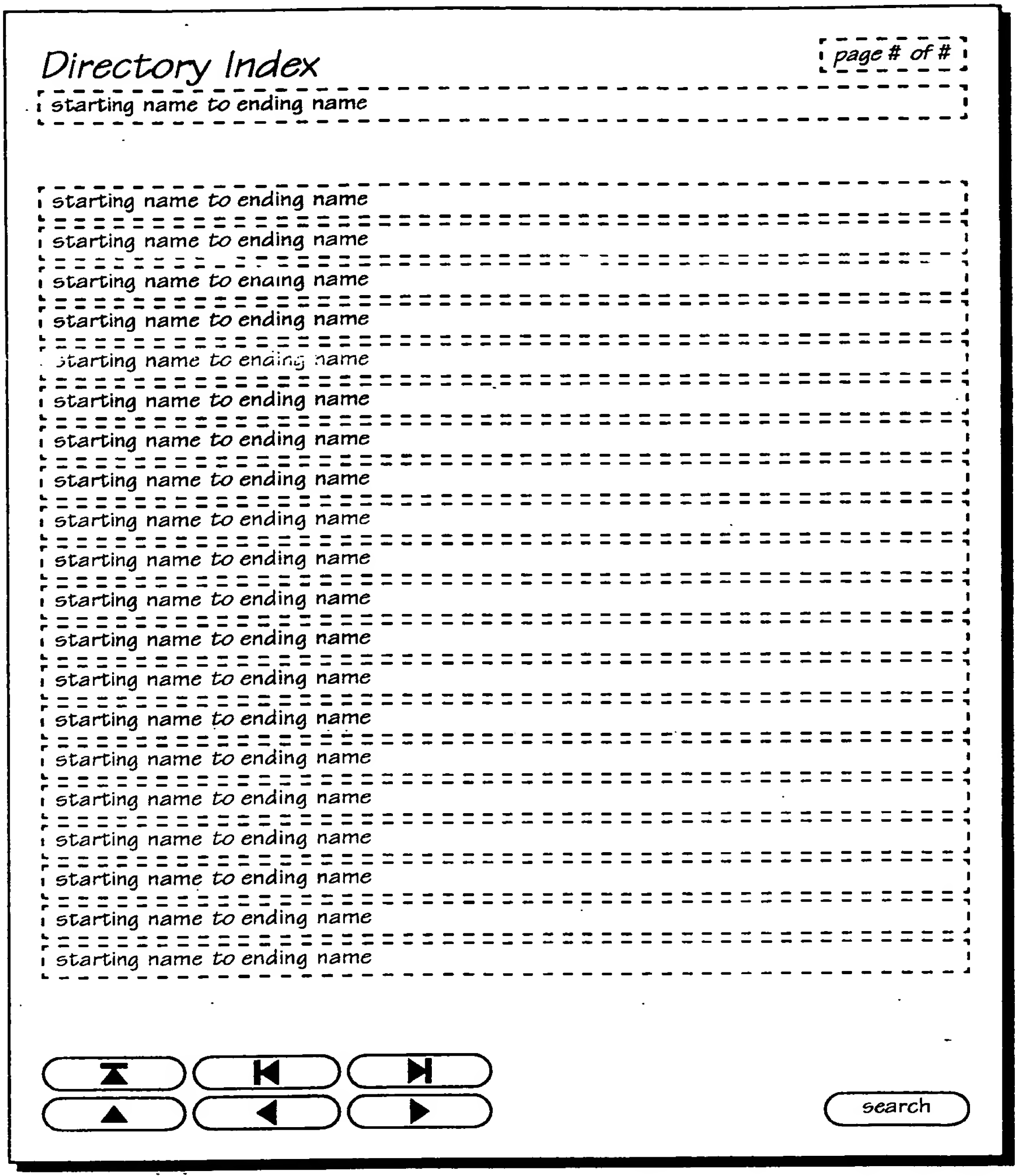


Figure 7. Generic directory index page

3 User Registration

Much of the Netpage System is predicated on the system's secure knowledge of user identity. This allows the system to:

- deliver personalized services to individual users independent of location
- support authenticated transactions
- support secure payments
- guarantee a high level of privacy

3.1 User Object Model

The user object model revolves around a uniquely identified user, as shown in the class diagram in Figure 8.

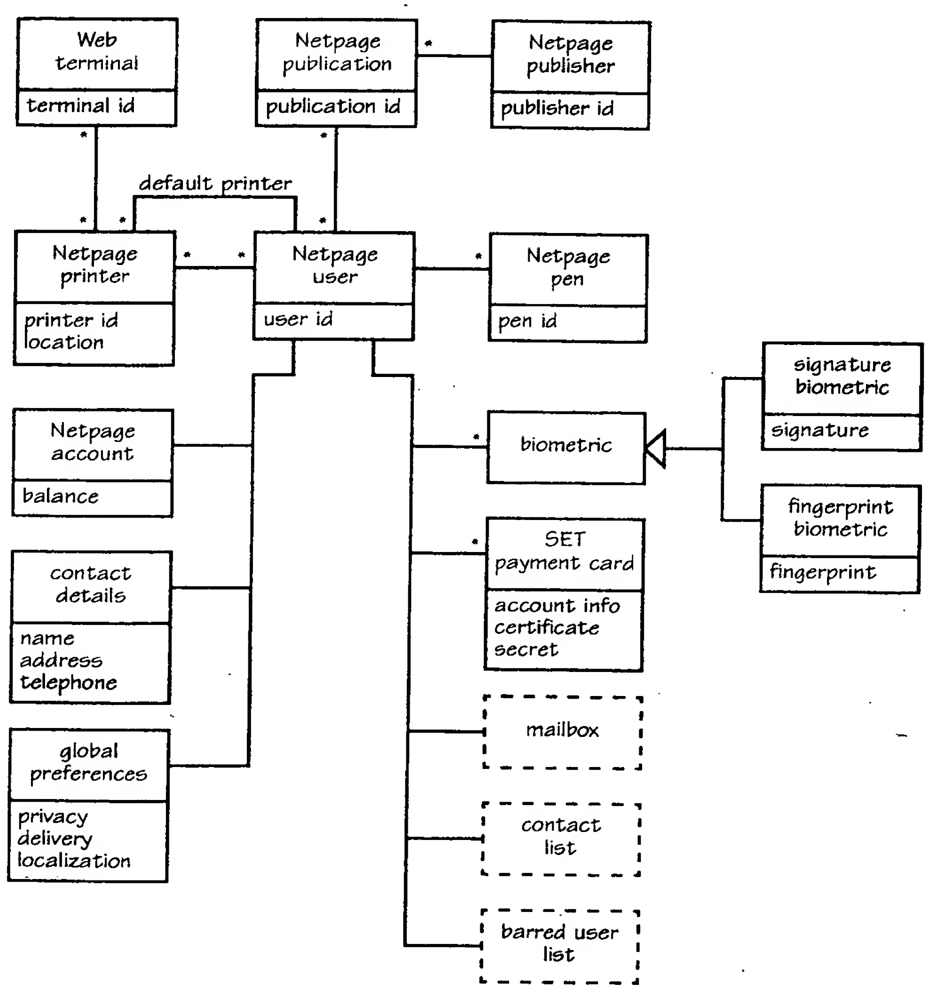


Figure 8. User class diagram

A user is authorized to use any number of printers, and printer may have any number of users. One of the user's printers is designated as the default printer, and subscriptions are

automatically delivered to the default printer. Pages requested ad hoc, on the other hand, are automatically delivered to the printer through which the user is interacting. Thus a publisher never records the id of a printer, but instead resolves the id when it is required.

When a user subscribes to a publication, the publisher is authorized to print to the user's default printer. This authorization can be revoked at any time by the user.

A Web terminal may be authorized to print to a Netpage Printer via the Netpage Network. This allows it to conveniently print Web and Netpage documents encountered while navigating the Web.

A user may have any number of pens, but each pen is linked to a particular user. Each of a user's pens may be used interchangeably.

Each user also has a Netpage account which is used to accumulate micro-transactions; contact details, including name, address, and telephone numbers; global preferences, including privacy, delivery, and localization settings; any number of biometric records, containing the user's encoded signature, fingerprint etc.; and any number of payment card accounts, with which e-commerce payments can be made.

The user's mailbox and contact list are described in Section 4.

3.2 User Registration User Interface

3.2.1 Help Page

The help page contains the following buttons which generate registration forms:

- add user
- add pen to user
- add pen to local user
- authorize user on printer
- authorize global user on printer
- authorize terminal on printer

3.2.2 User Registration

A user must be registered with the Netpage Network, own at least one registered Netpage Pen, and be authorized on at least one Netpage Printer, before being able to meaningfully interact with the Netpage Network.

The <add user> button on the help page generates a **user registration** form (Figure 10) which captures the user's contact details and privacy preferences.

The pen used to submit the user registration form, if previously unregistered, is automatically linked to the new user. The user is automatically authorized to use the printer through which the registration takes place.

A subset of the users authorized to use a printer are designated as administrators, and only they may authorize the registration of additional users and pens on the printer.

By default, every new user is an administrator on a particular printer until a printer-specific privileges setting is enabled (via the help page). Thus a new user can be registered

without having to be authorized by an administrator's pen until the privileges setting is enabled. This setting may never need to be enabled in a domestic environment, but may commonly be enabled in a corporate environment. User-specific privileges settings only appear on the registration form if the printer-specific privileges setting is enabled.

When privileges are enabled, the user registration form must be authorized by an administrator's pen. This is done by initialling the registration form in the appropriate place.

The user registration user interface flow is illustrated in Figure 9.

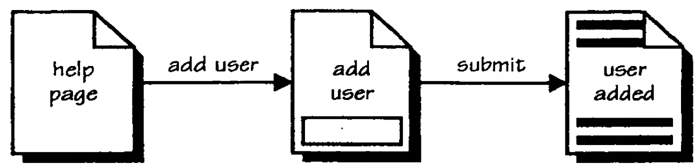


Figure 9. User registration user interface flow

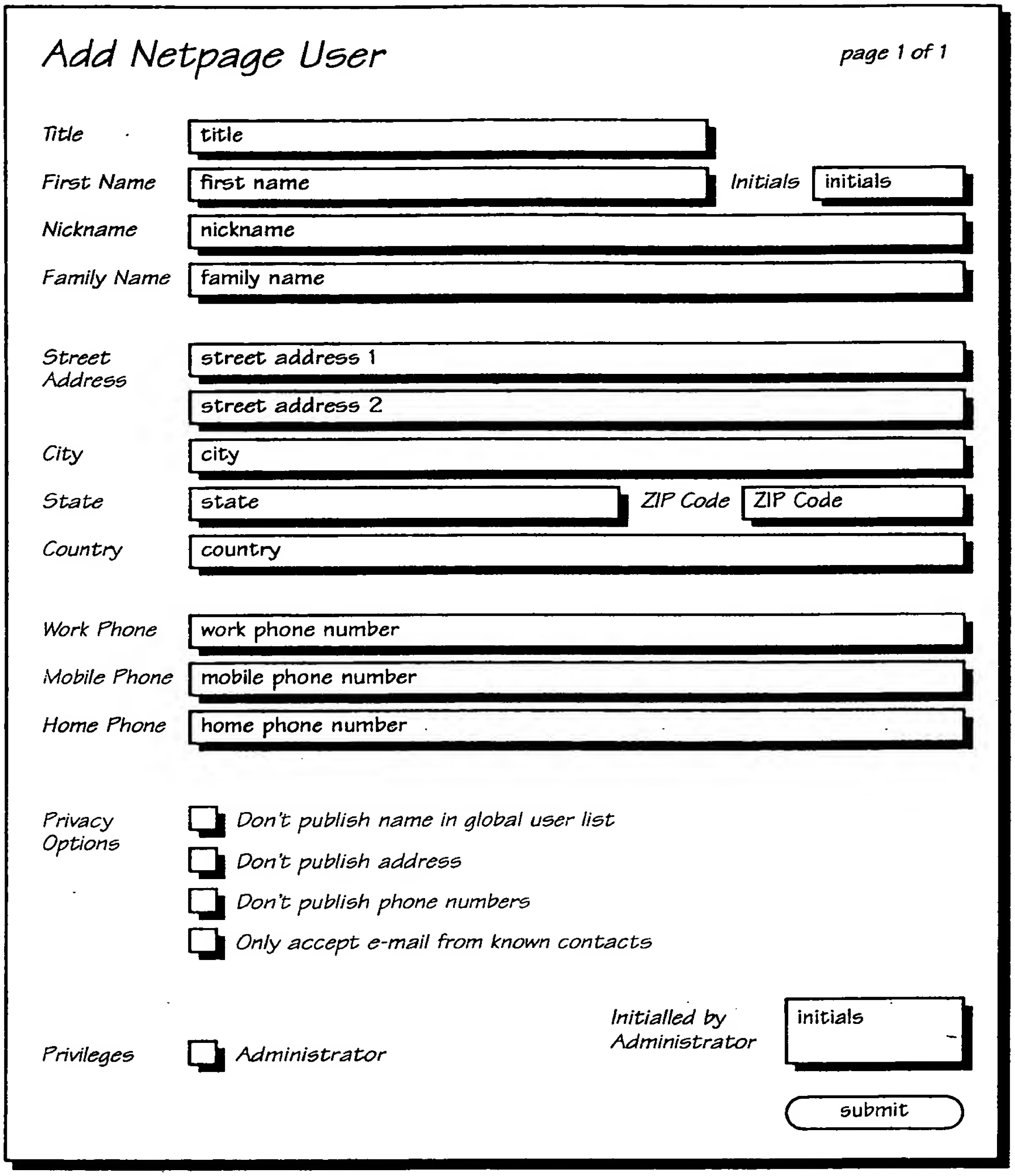


Figure 10. User registration page

3.2.3 Pen Registration

Additional Netpage Pens may be linked to existing users.

The <add pen to user> button on the help page generates an **pen registration** form (Figure 12). The form contains name and nickname fields and a <submit> button. If the form is requested using the new pen, i.e. the pen to be registered, then the form is blank and must be filled in with sufficient name details to identity the user. If the form is requested using an existing pen, then the form is automatically filled in with the details of the existing pen's owner.

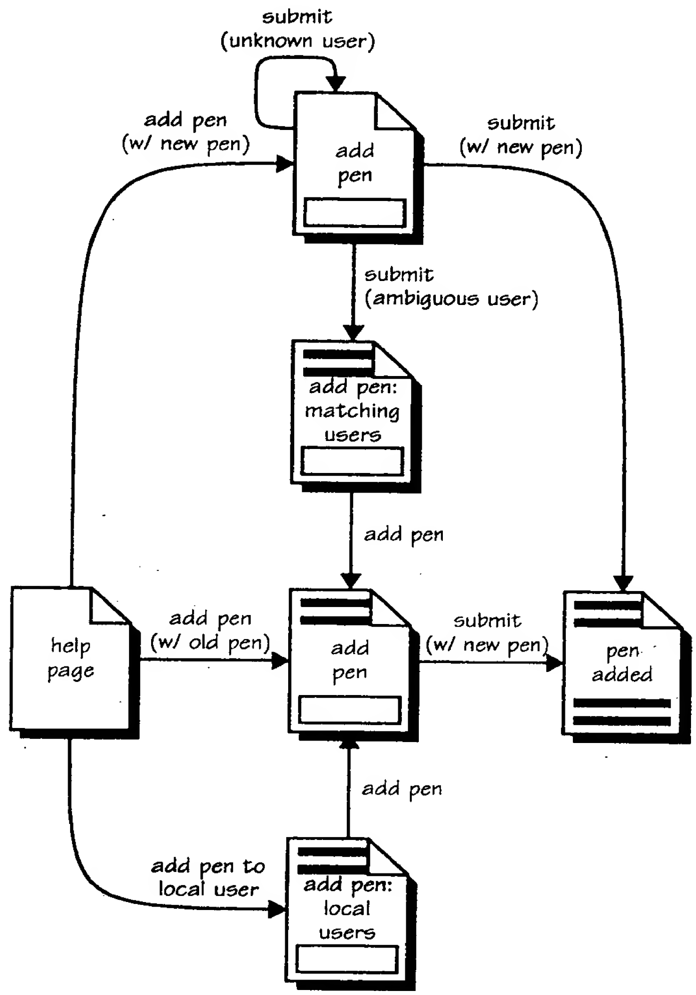


Figure 11. Pen registration user interface flow

When the form is submitted, the name information is matched against the list of local users, i.e. users authorized to use the printer. Partial name information can be entered, such as just a first name or a nickname. If the name information is ambiguous, the system generates a list of matching users (Figure 13) with an <add pen> button next to each.

The user must submit the pen registration form using the new pen. This links the new pen to the user. The form cannot be submitted using a pen which is already registered.

The <add pen to local user> button on the help page generates a list of all users authorized to use the printer (Figure 14), with an <add pen> button next to each. The local user list is only available to administrators.

When privileges are enabled, the pen registration form must be authorized by an administrator's pen. This is done by initialling the registration form in the appropriate place.

The pen registration user interface flow is illustrated in Figure 11.

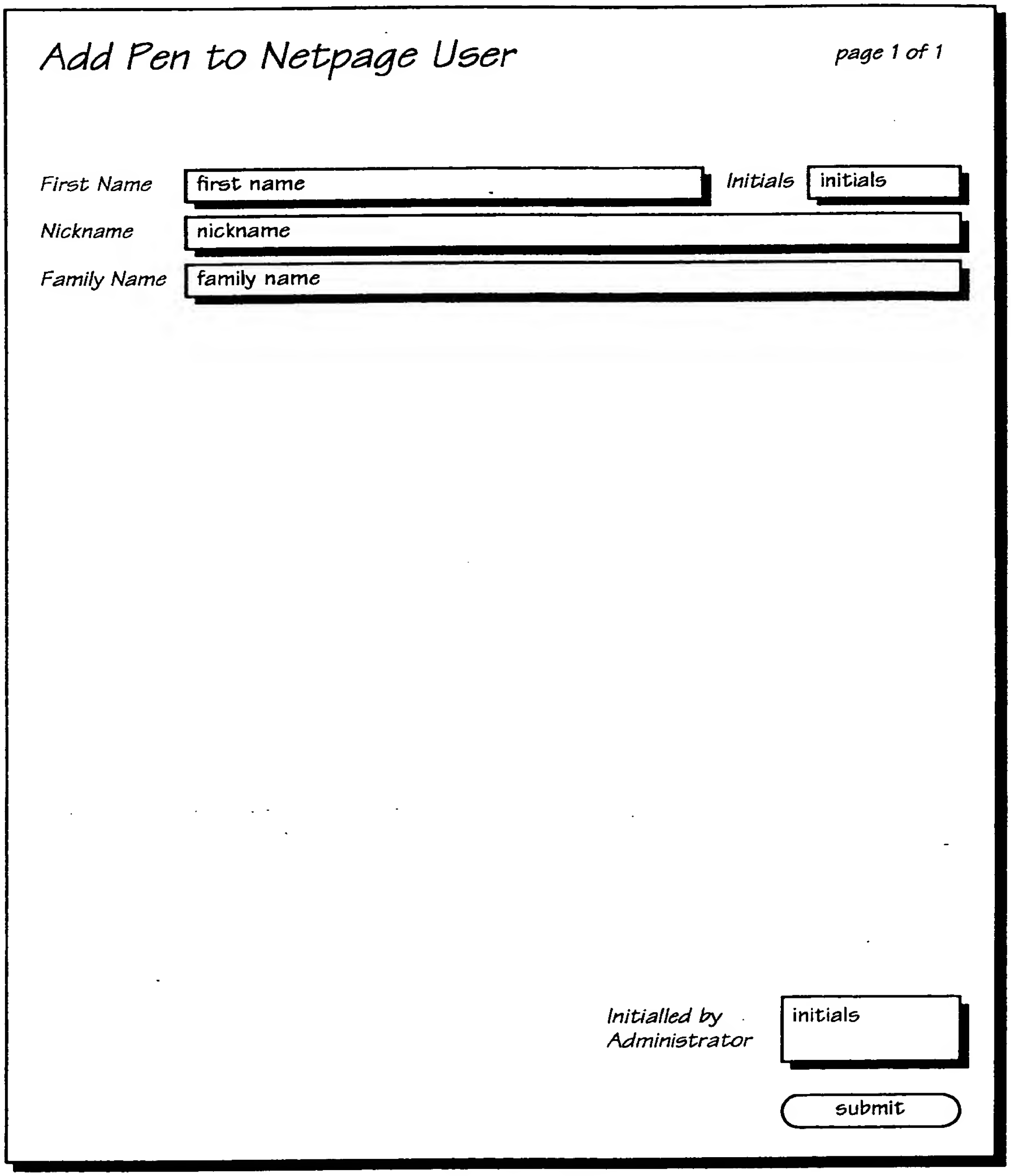


Figure 12. Pen registration page

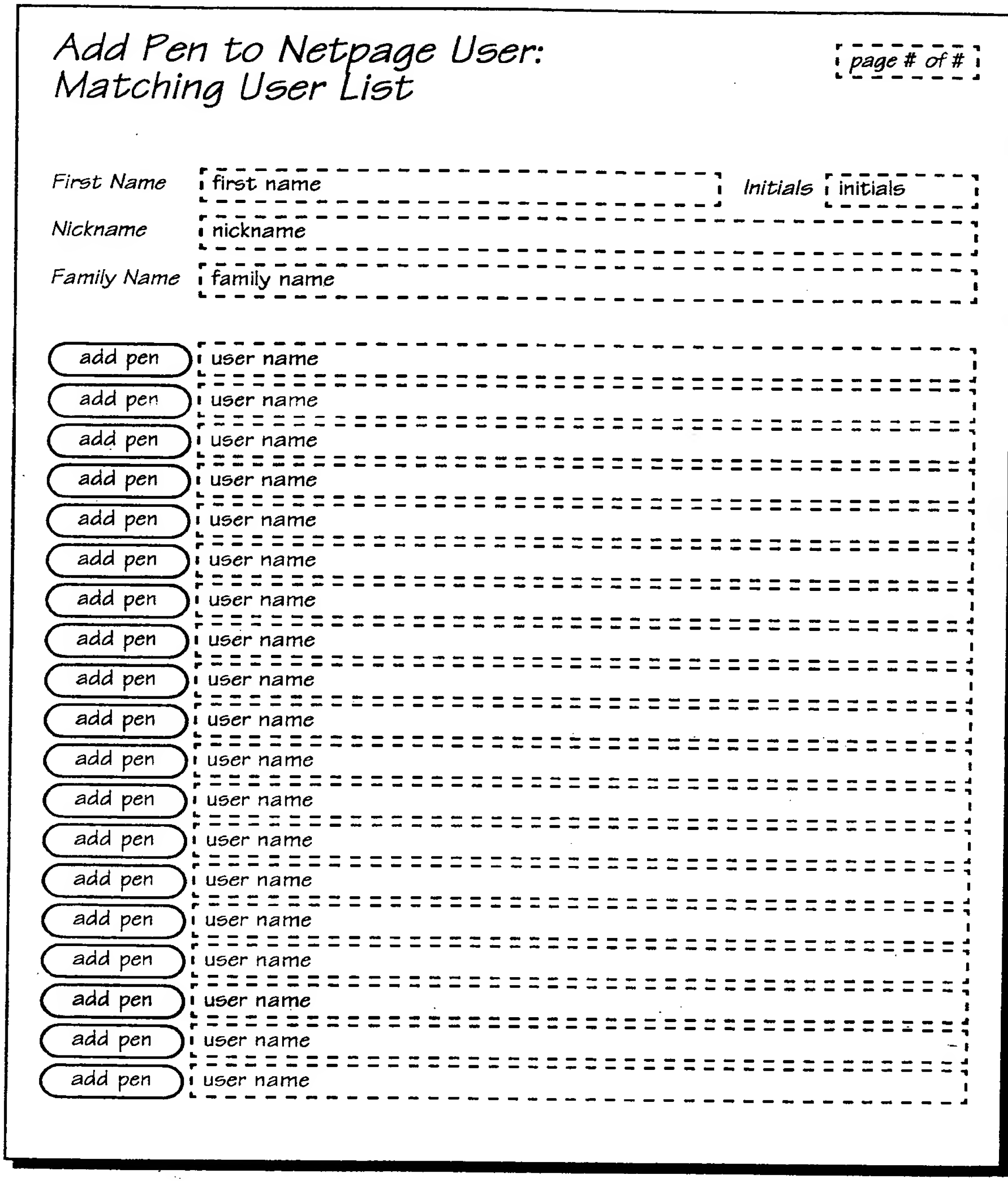


Figure 13. Pen registration - matching users page

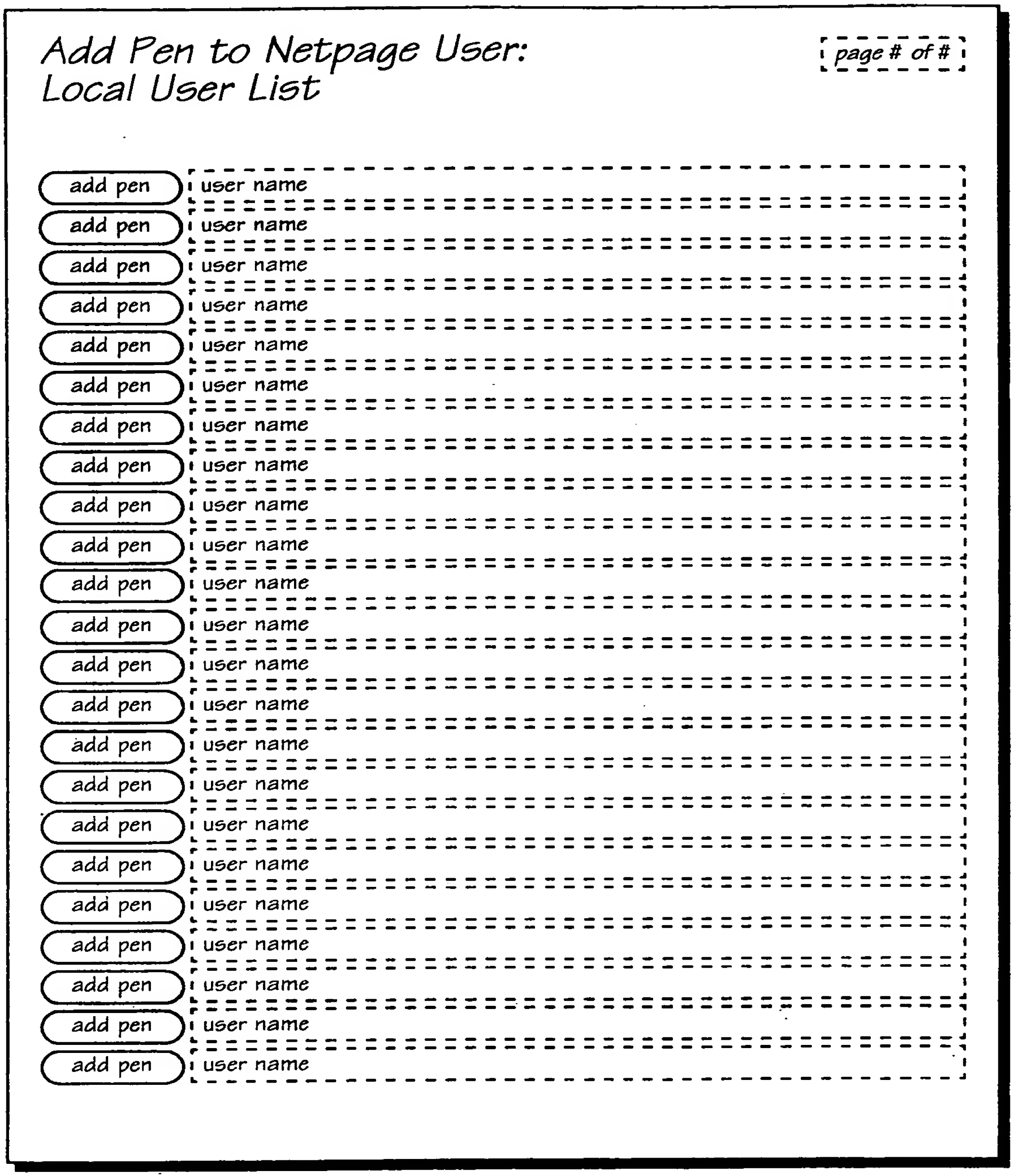


Figure 14. Pen registration - local users page

3.2.4 Authorize User on Printer

A user may be authorized to use additional Netpage Printers. Once authorized to use a particular printer, the user may interact with the printer using any pen linked to the user.

The <authorize user on printer> button on the help page generates a **user authorization** form (Figure 16). The registration form contains name and nickname fields and a <submit> button. If the form is requested using a pen owned by a local user, then the form is blank and must be filled in with sufficient name details to identity the new user. If the form is requested using the new user's pen, then the form is automatically filled in with the details of the new user.

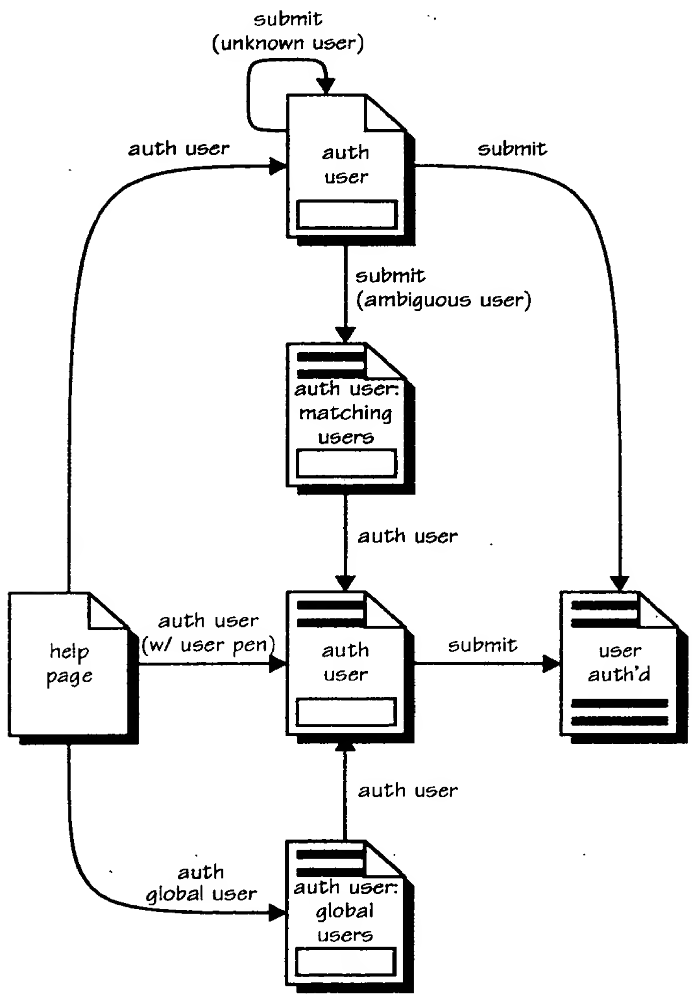


Figure 15. User authorization user interface flow

When the form is submitted, the name information is matched against the global user list. Partial name information can be entered, such as just a first name or a nickname. If the name information is ambiguous, the system generates a list of matching users (Figure 17) with an <authorize user> button next to each.

The <authorize global user> button on the help page generates a list of all users on the network (Figure 18), with an <authorize user> button next to each. The global user list is available to all users, but only contains users who have elected to appear.

When privileges are enabled, the user authorization form must be authorized by an administrator's pen. This is done by initialling the authorization form in the appropriate place.

The user authorization user interface flow is illustrated in Figure 15.

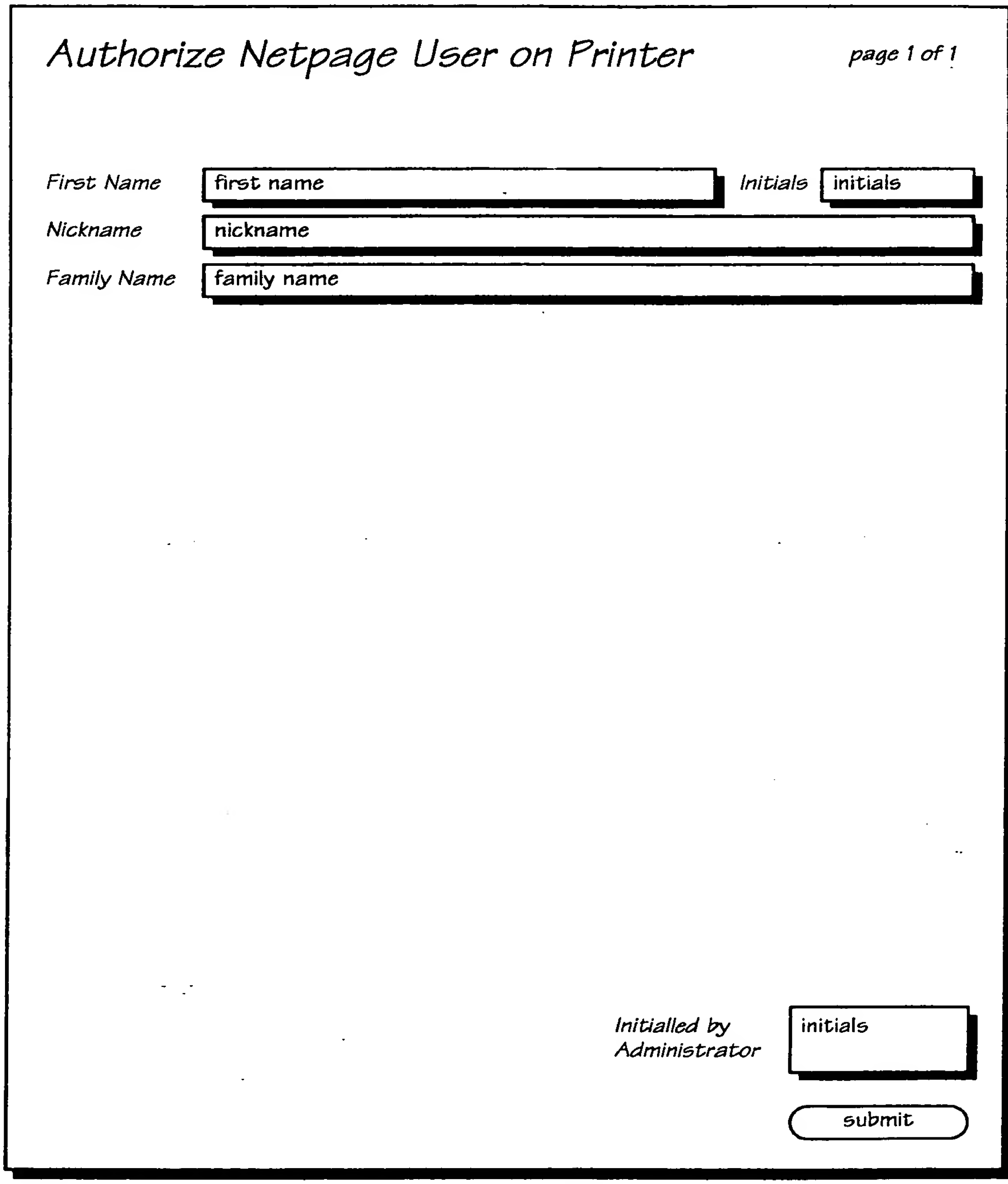


Figure 16. User authorization page

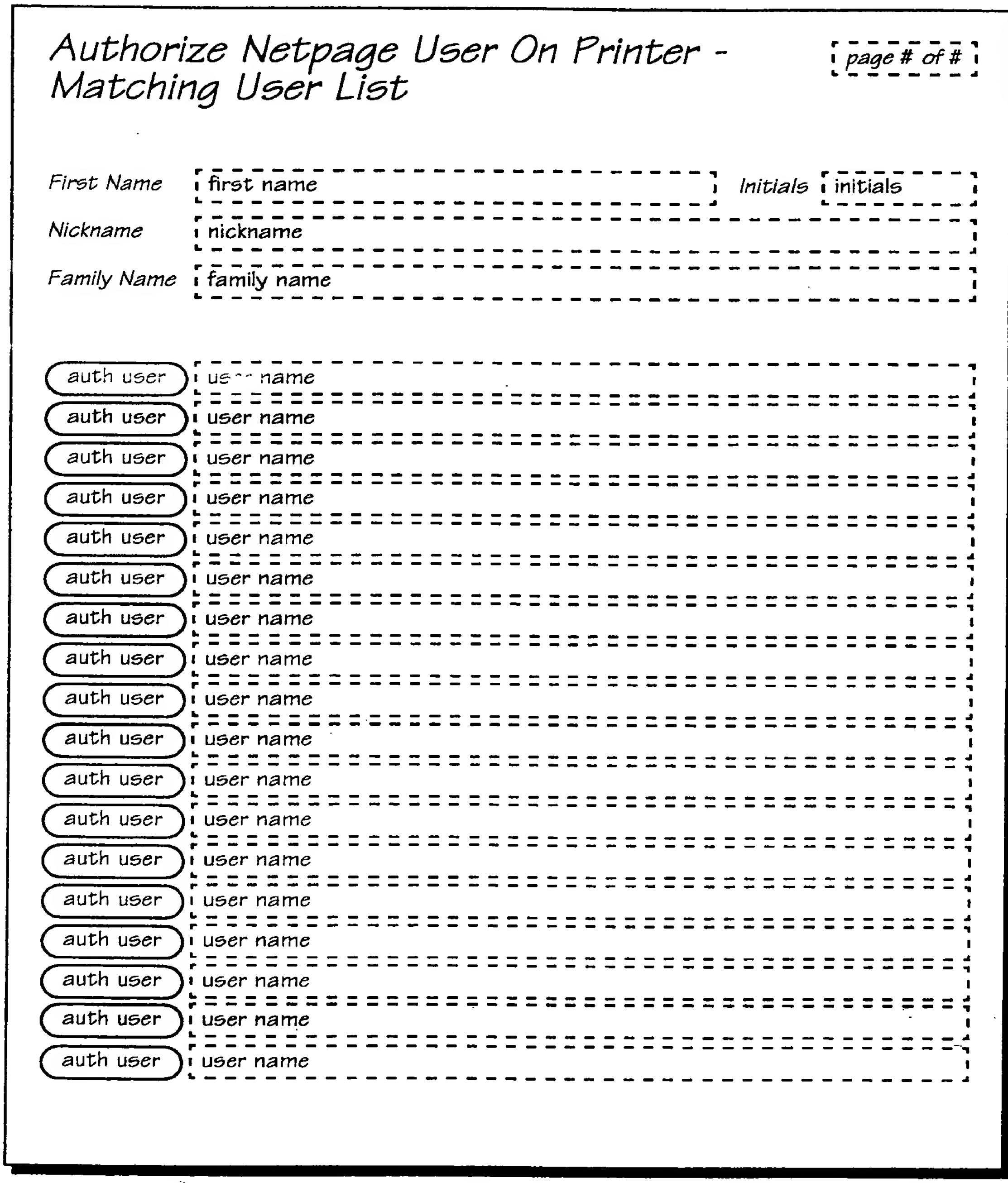


Figure 17. User authorization - matching users page

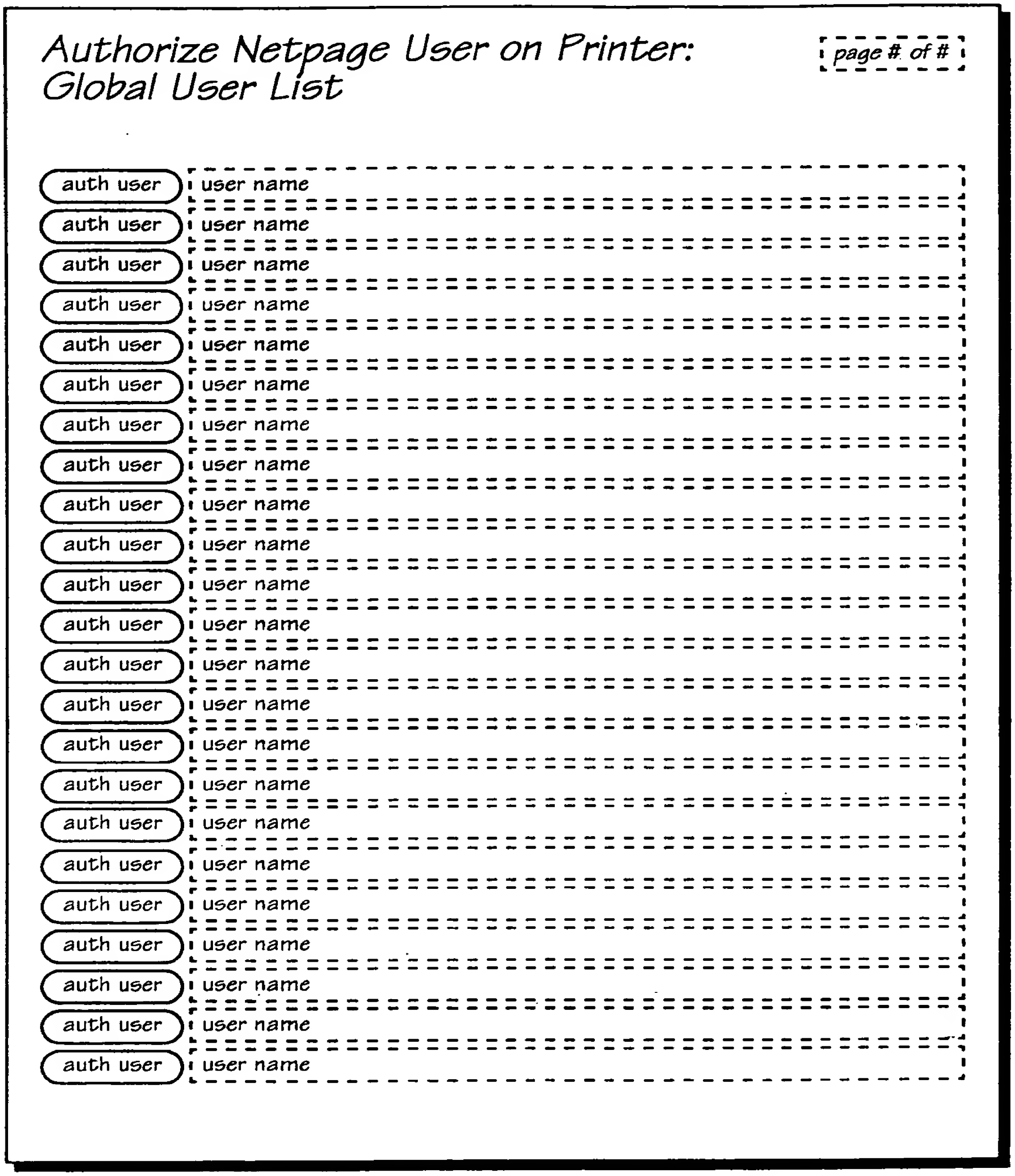


Figure 18. User authorization - global users page

3.2.5 Authorize Terminal on Printer

The <authorize terminal on printer> button on the help page produces a terminal authorization token. The token is printed with information which allows the user of the Web terminal to register the Web terminal with the Netpage Network, allowing it to print to the Netpage Printer identified by the token.

An administrator may revoke a terminal's authorization at any time. The <authorize terminal on printer> button is also only available to administrators.

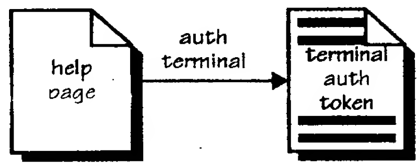


Figure 19. Terminal authorization user interface flow

4 Electronic Mail

Netpage electronic mail (e-mail) provides a messaging service between Netpage users. It also supports message exchange with Internet e-mail users, and by extension, users of other e-mail systems interconnected with the Internet, such as corporate e-mail systems.

Each Netpage user has a unique id within the Netpage System, and is also known to the system by their name, which may or may not be unique. Netpage e-mail is usually addressed by selecting a name from a list, for example from the global list of Netpage users or from a particular user's list of contacts. The user's nickname or alias helps disambiguate similar names. A Netpage user usually doesn't have to know or specify another Netpage user's unique id.

When a Netpage user wants to send e-mail to an Internet user, they must specify an Internet e-mail address. Similarly, when an Internet user wants to send e-mail to a Netpage user, they must specify the Netpage e-mail address. A Netpage e-mail address takes the form `<userid>@netpage.net`.

E-mail is a core service of the Netpage System.

4.1 E-MAIL OBJECT MODEL

4.1.1 E-mail User

The e-mail object model revolves around an e-mail user. An e-mail user is either a Netpage user or an Internet user.

The e-mail user class diagram is shown in Figure 20.

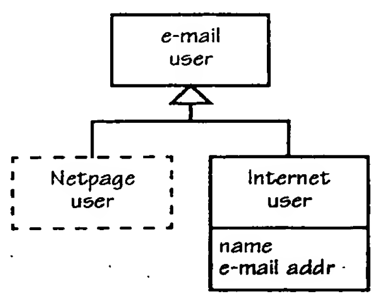


Figure 20. E-mail user class diagram

4.1.2 E-mail

The e-mail itself consists of a number of pages. Each page corresponds to a Netpage. The page structure is logical because the e-mail is delivered in the same form that it is composed, and it is composed by the sender page by page.

Each page contains digital ink, together with any number of attachments. An attachment is printed at its point of insertion, and may overflow onto the following page(s).

The sender of the e-mail is an e-mail user, as are any number of recipients and copy recipients (CCs). Blind copy recipients (BCCs) are not modelled here for clarity, but can obviously be trivially accommodated.

The e-mail has a explicit subject which is always converted from digital ink to text to support interoperation with e-mail systems which don't directly support digital ink, such as e-mail systems on the Internet, and to streamline the presentation of mailbox contents within the Netpage System itself.

The e-mail has a high priority flag which allows the both sender and the recipients to control the timeliness of its delivery.

The e-mail class diagram is shown in Figure 21.

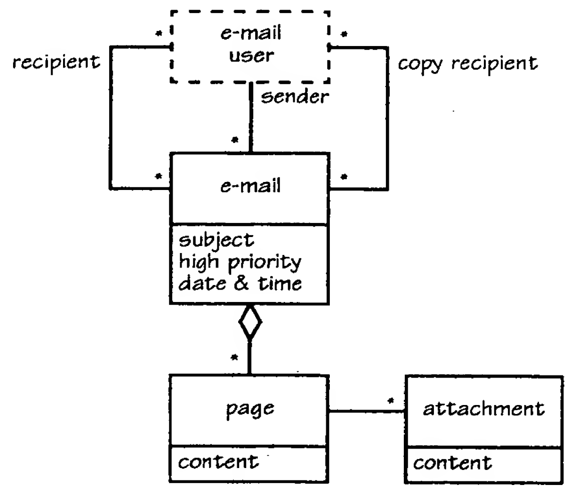


Figure 21. E-mail class diagram

4.1.3 Mailbox

Incoming e-mails may be accumulated in the user's mailbox, printed on the user's default printer, incorporated into the user's daily newspaper, or any combination of these. The user can select, for example, to print high-priority e-mails immediately on receipt but to hold low-priority e-mails.

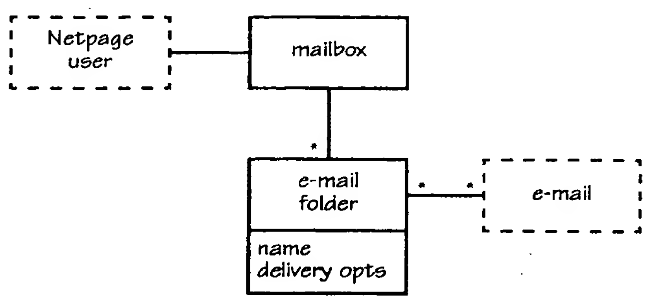


Figure 22. Mailbox class diagram

The user may create any number of named folders within the mailbox, and either manually copy and move received e-mails between folders, or associate a folder with one or more e-mail contacts (including e-mail groups) to have e-mail received from those contacts automatically placed in the folder. E-mail received from contacts not associated with a folder is placed in a predefined "inbox" folder. All e-mail sent by the user is placed in a predefined "sent e-mail" folder for future reference.

Each folder has three delivery options: "print all e-mail", "print high-priority e-mail", and "delete e-mail once printed". If a print option is selected, e-mail messages with the corresponding priority are printed on the user's default printer immediately on receipt. If the delete option is selected, e-mail messages are deleted from the e-mail folder once printed. Otherwise e-mail messages are held indefinitely in the e-mail folder, i.e. until manually deleted by the user.

The mailbox class diagram is shown in Figure 22.

4.1.4 Contact List

Each user has a list of contacts. The contact list provides a more convenient basis for selecting e-mail recipients than the global list of users, particularly since most user's contact lists will fit on a single double-sided printed page. The contact list also provides a basis for ignoring unsolicited e-mail. A privacy option allows a user to ignore e-mail not sent by a member of the contact list.

A user may create any number of contact groups within the contact list, and treat a group as a single contact for the purposes of addressing outgoing e-mail and controlling the delivery of incoming e-mail.

Groups may themselves contain groups, and both individual contacts and groups may be members of multiple groups as well as the top-level contact list itself.

The contact list class diagram is shown in Figure 23.

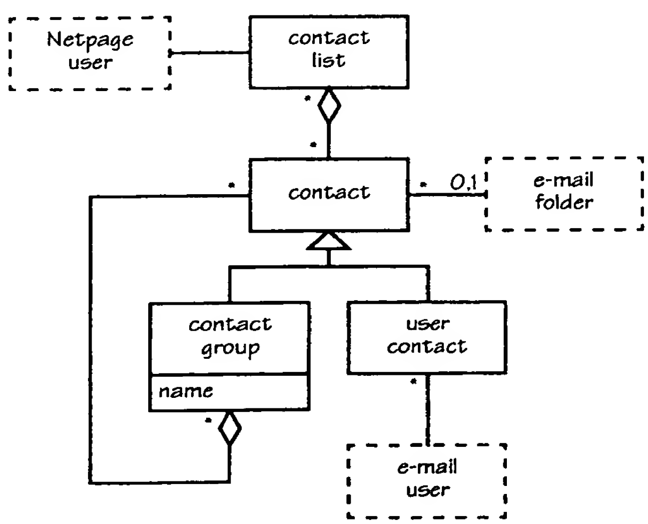


Figure 23. Contact list class diagram

4.1.5 Barred User List

Rather than accepting e-mail only from known contacts, a user may choose to bar individual users. The barred user list records individuals from which the user refuses to accept e-mail.

The barred user list class diagram is shown in Figure 24.

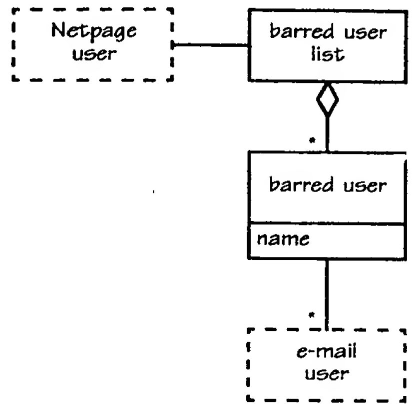


Figure 24. Barred user list class diagram

4.2 E-MAIL USER INTERFACE

4.2.1 Send E-mail

E-mail is sent using the **outgoing e-mail** form (see Figures 27 and 28). The e-mail form is always printed with the sender's name already specified, since the sender's identity is known from the pen used to request the e-mail form. If the e-mail form is requested by pressing a user-specific <e-mail> button on some page, then the e-mail form is printed with the recipient's name already specified. On the other hand, if the e-mail form is requested from the help page, then the recipient is unknown and is left blank on the form. The user must specify at least one recipient before the e-mail can be sent.

The user can also pre-address an e-mail form by pressing the <e-mail to contacts> button on the help page. This elicits an **add recipients** form (Figure 29), which lists the user's contacts. Each contact has a <To> and a <CC> checkbox as well as an <e-mail> button. The e-mail button elicits an e-mail form addressed to that user. The <e-mail selected> button at the bottom of the page elicits an e-mail addressed to all the users whose <To> or <CC> checkboxes have been checked.

The e-mail form initially consists of a double-sided page. The front (Figure 27) contains fields for the names of recipients and copy recipients, for the subject, and for the body of the e-mail. The back (Figure 28) contains a field for continuing the body of the e-mail. Any recipient names (or addresses) written by hand are converted from digital ink to text for lookup purposes. The subject is also converted, for presentation purposes, as discussed earlier. The body is retained as digital ink, to allow handwritten text and diagrams etc. to be delivered in a uniform and expressive manner.

The <add page> button at the bottom of every page of the e-mail form adds another double-sided page to the e-mail form. Only the additional page is printed, not the entire form,

but the additional page is logically linked to the original e-mail. The <discard page> button at the bottom of every page discards the corresponding page from the e-mail (but not both sides of the double-sided page). The entire e-mail is reprinted with the discarded page removed and subsequent pages, if any, moved to close the gap. A blank page added to make the page count even, if necessary.

Each printed e-mail form corresponds to a separate e-mail instance, the uniqueness of which is indicated by the sender's name, date and time printed at the top of every page. Once an e-mail has been sent, it can't be sent again. A copy, however, can be easily made, edited, and sent. The standard <print> button on every page of the e-mail form prints another copy of the e-mail form, corresponding to a new e-mail instance. Both the original and the copy can be edited further and sent independently.

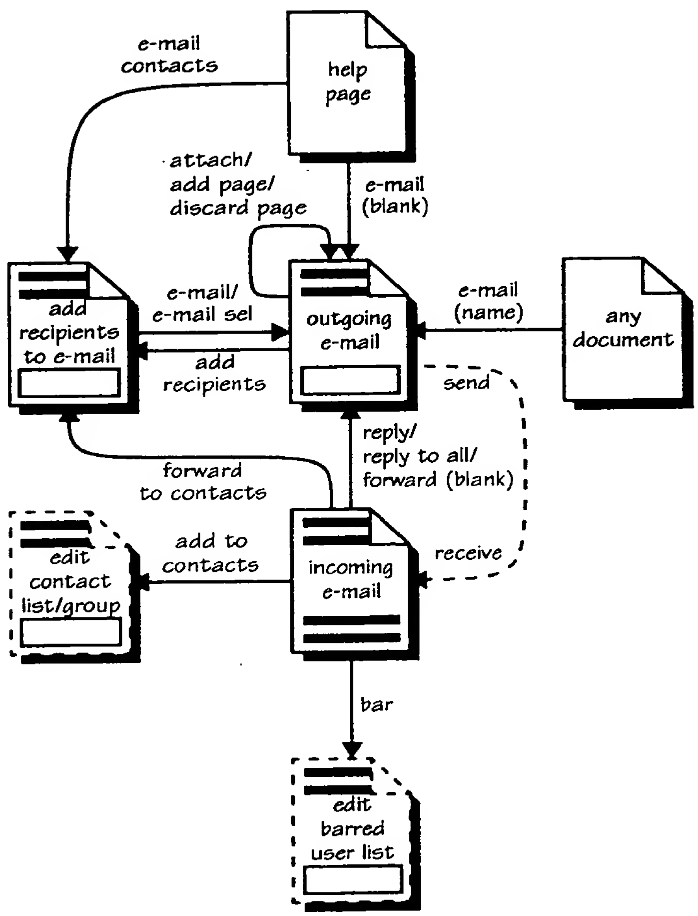


Figure 25. E-mail user interface flow

Text in the <recipients>, <copy recipients> and <subject> can be struck out with the pen to remove recipients, subject text, etc. The <print> button produces a copy of the e-mail form with the struck-out text removed.

The <attach> button at the bottom of every page attaches the current selection at the current end of the e-mail body. The entire e-mail is reprinted with the attachment included.

Additional pages are automatically added to the e-mail to accommodate the attachment. The attachment can consist of anything which can be selected on any Netpage.

The <add recipients> button adjacent to the <recipients> and <copy recipients> fields at the top of the first page of the e-mail form elicits an add recipients form (Figure 29), with the <subject> field reflecting the subject of the e-mail form. The <e-mail> and <e-mail selected> buttons on the add recipients form elicits a copy of the e-mail form with the additional recipients and copy recipients added.

The <send> button at the bottom of every page sends the entire e-mail. If the name of any recipient is unknown to the system, then the e-mail form is reprinted with the offending name printed in red, together with an error message indicating the problem.

If the name of any recipient is ambiguous, then a list of matching users is printed, each with a checkbox allowing it to be selected. The <e-mail> button at the bottom of the form reprints the e-mail form with the recipient names suitably updated.

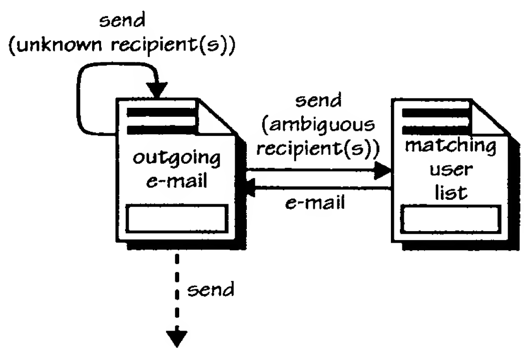


Figure 26. E-mail recipient exception user interface flow

The overall e-mail user interface flow is illustrated in Figure 25. The e-mail recipient exception user interface flow is illustrated in Figure 26.

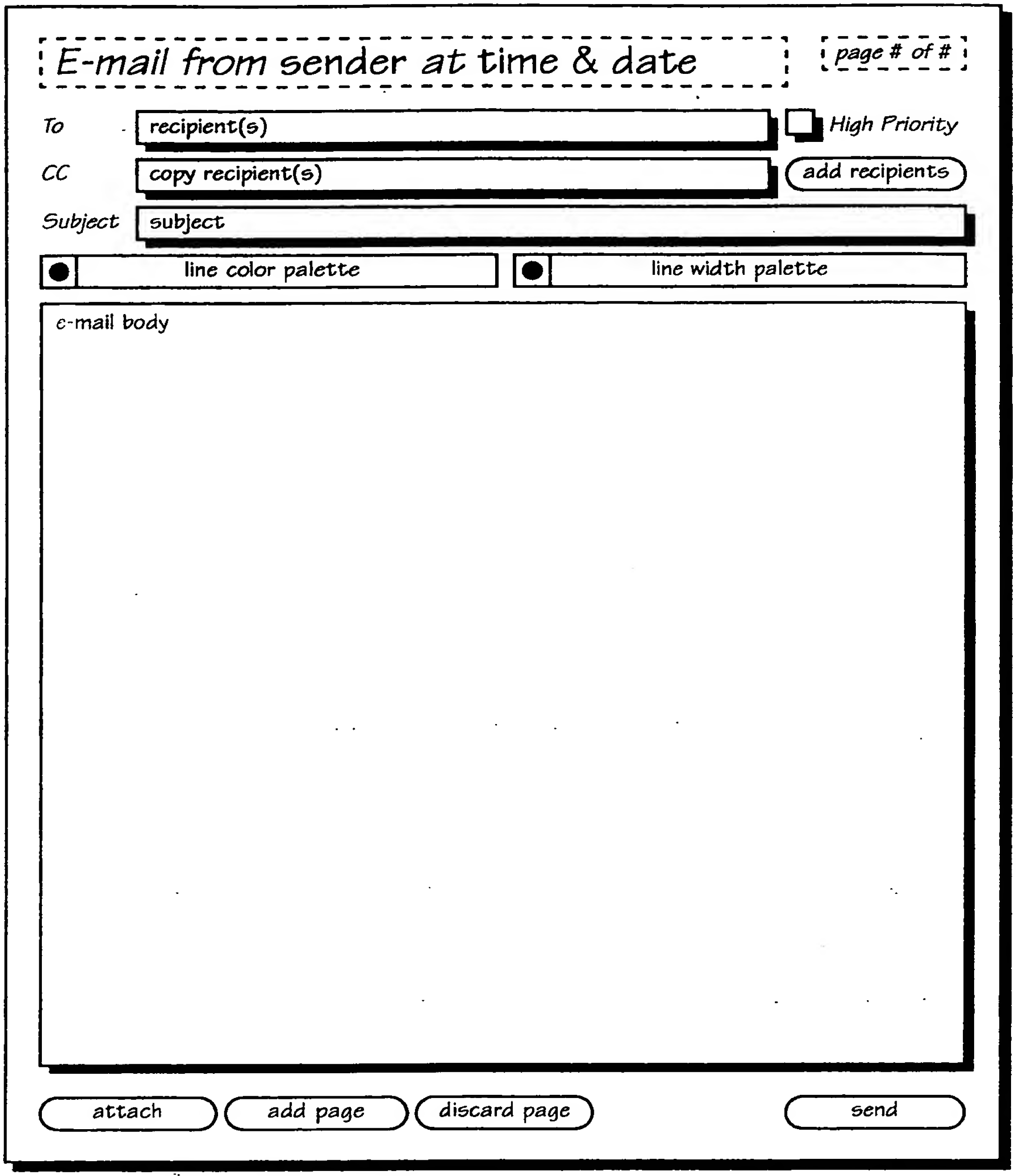


Figure 27. Outgoing e-mail first page

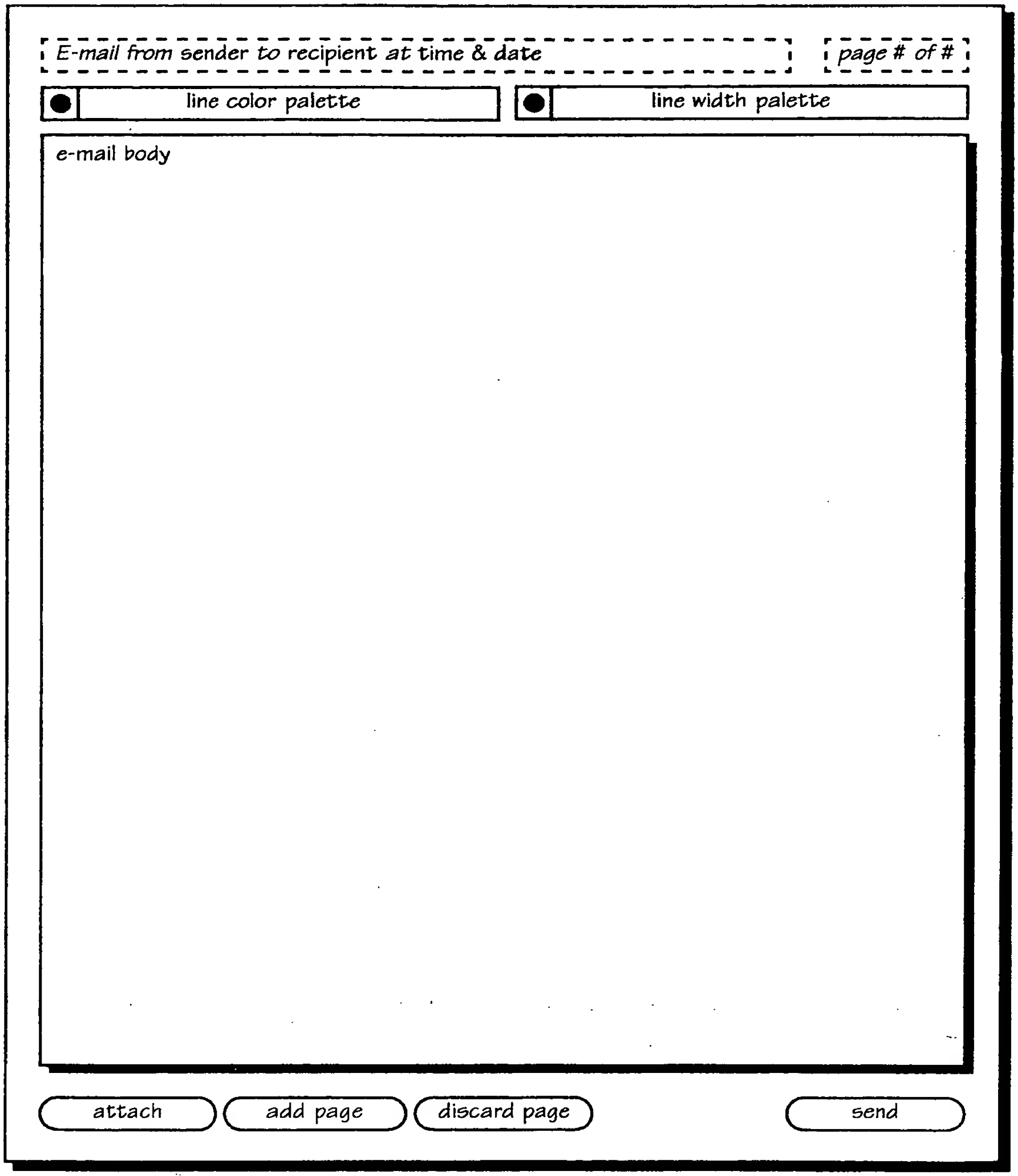


Figure 28. Outgoing e-mail second or subsequent page

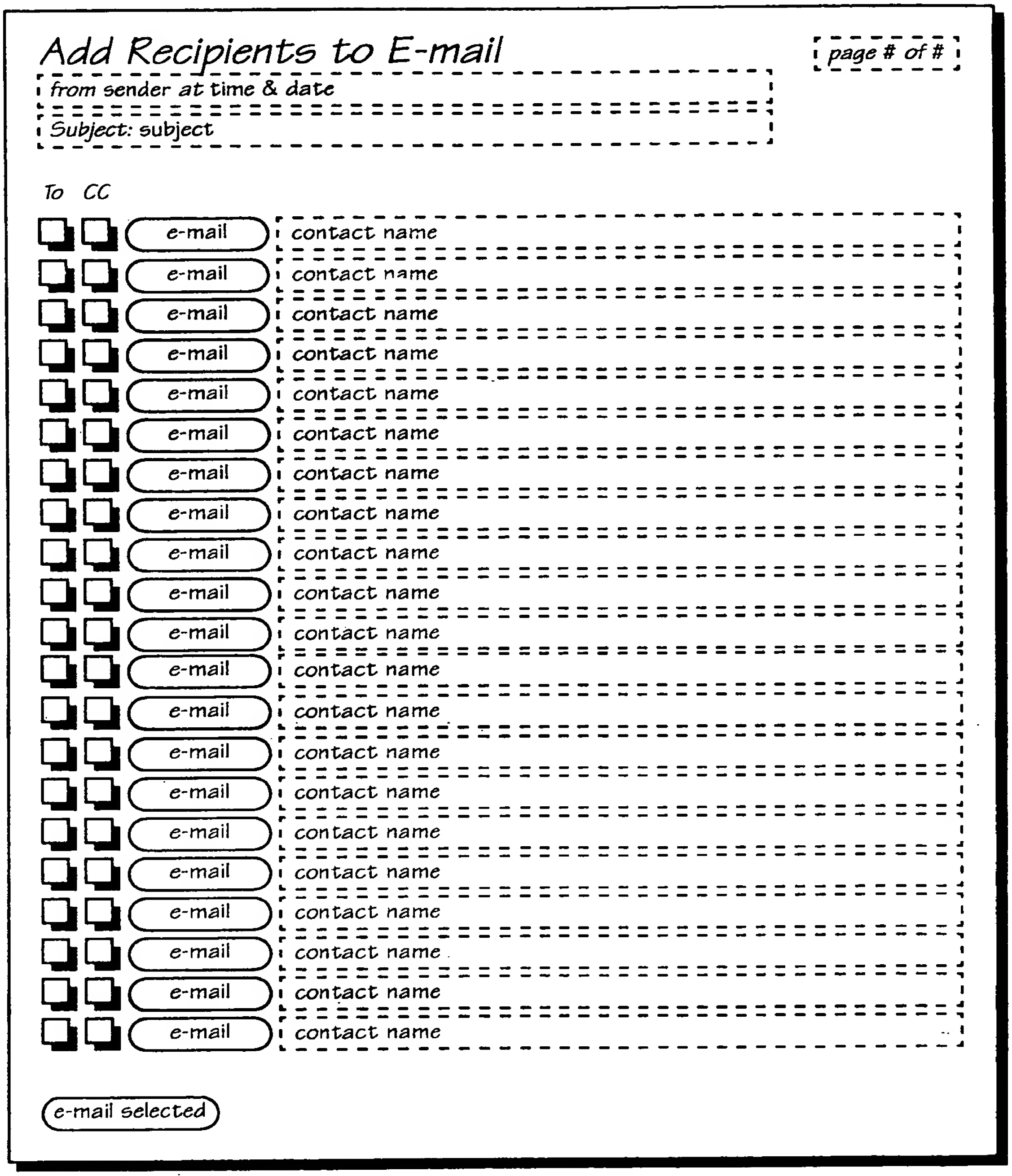


Figure 29. Add recipients page

4.2.2 Receive E-mail

E-mail is received in the form of an **incoming e-mail** document (Figures 30 and 31), which has the same page structure and content as the corresponding outgoing e-mail form.

The <reply> button at the bottom of every page of the e-mail document produces an outgoing e-mail form (Figures 27 and 28) addressed to the sender of the incoming e-mail, and with the subject reflecting the subject of the incoming e-mail, but with "In reply to:" added as a prefix.

The <reply to all> button produces an outgoing e-mail form addressed to the sender of the incoming e-mail, as well as to all of its recipients and copy recipients.

The <forward> button produces an outgoing e-mail form with no recipient, and with the subject reflecting the subject of the incoming e-mail, but with "Forwarded:" added as a prefix. The body of the incoming e-mail is also copied to the body of the outgoing e-mail.

The <forward to contacts> creates, but doesn't print, an outgoing e-mail in the same way as the <forward> button, and then implicitly invokes the <add recipients> command on it. This elicits an add recipients form in the usual way (Figure 29), with the <subject> field reflecting the subject of the forwarded e-mail.

The <add to contacts> button adds the sender to the recipient's list of contacts and produces an **edit contacts** form reflecting the updated contact list. Contact list editing is described in more detail in Section 4.2.3 on page 34.

The <bar> button adds the sender to the recipient's list of barred users and produces an **edit barred users** form reflecting the updated barred user list. Barred user list editing is described in more detail in Section 4.2.5 on page 43.

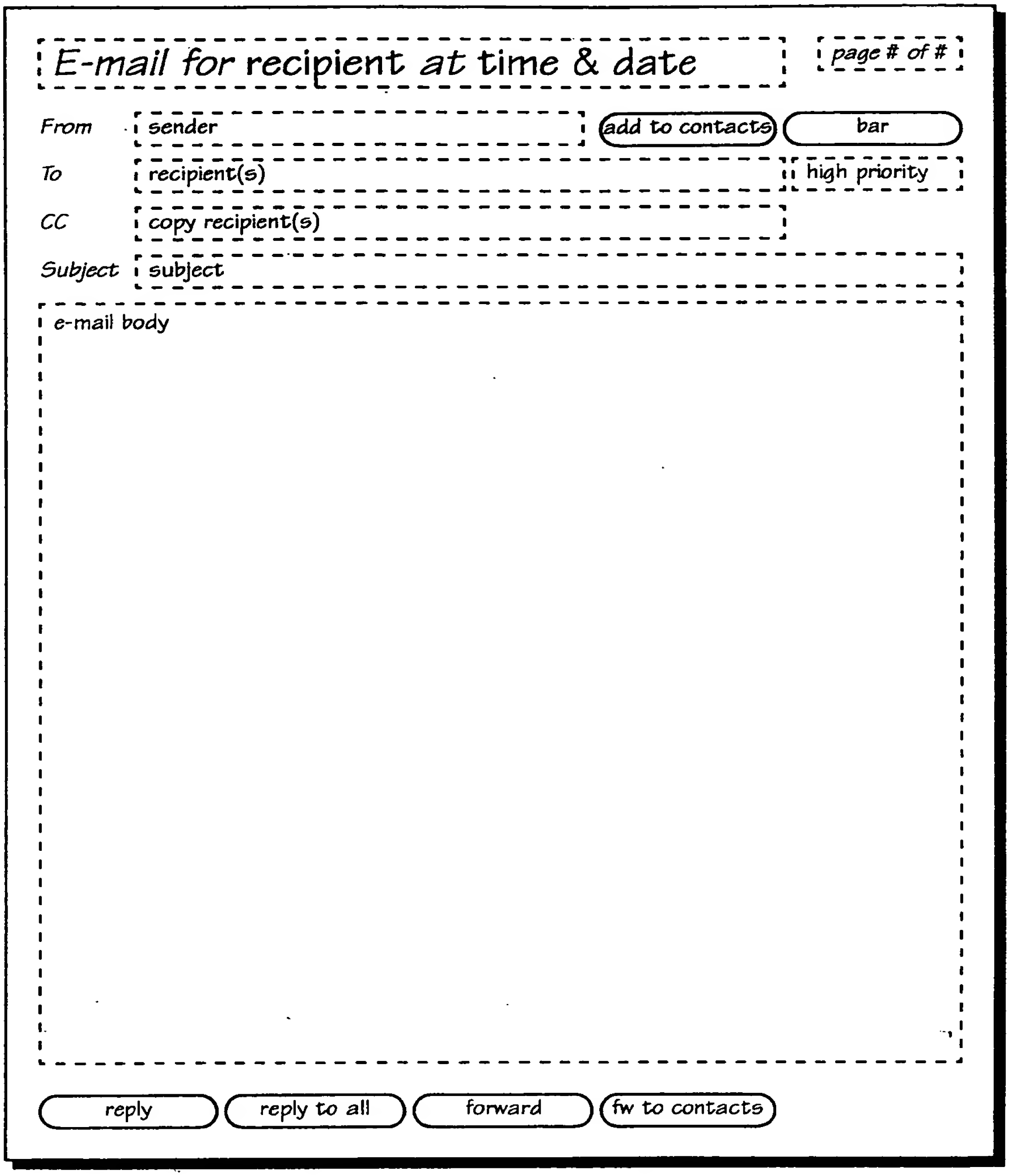


Figure 30. Incoming e-mail first page

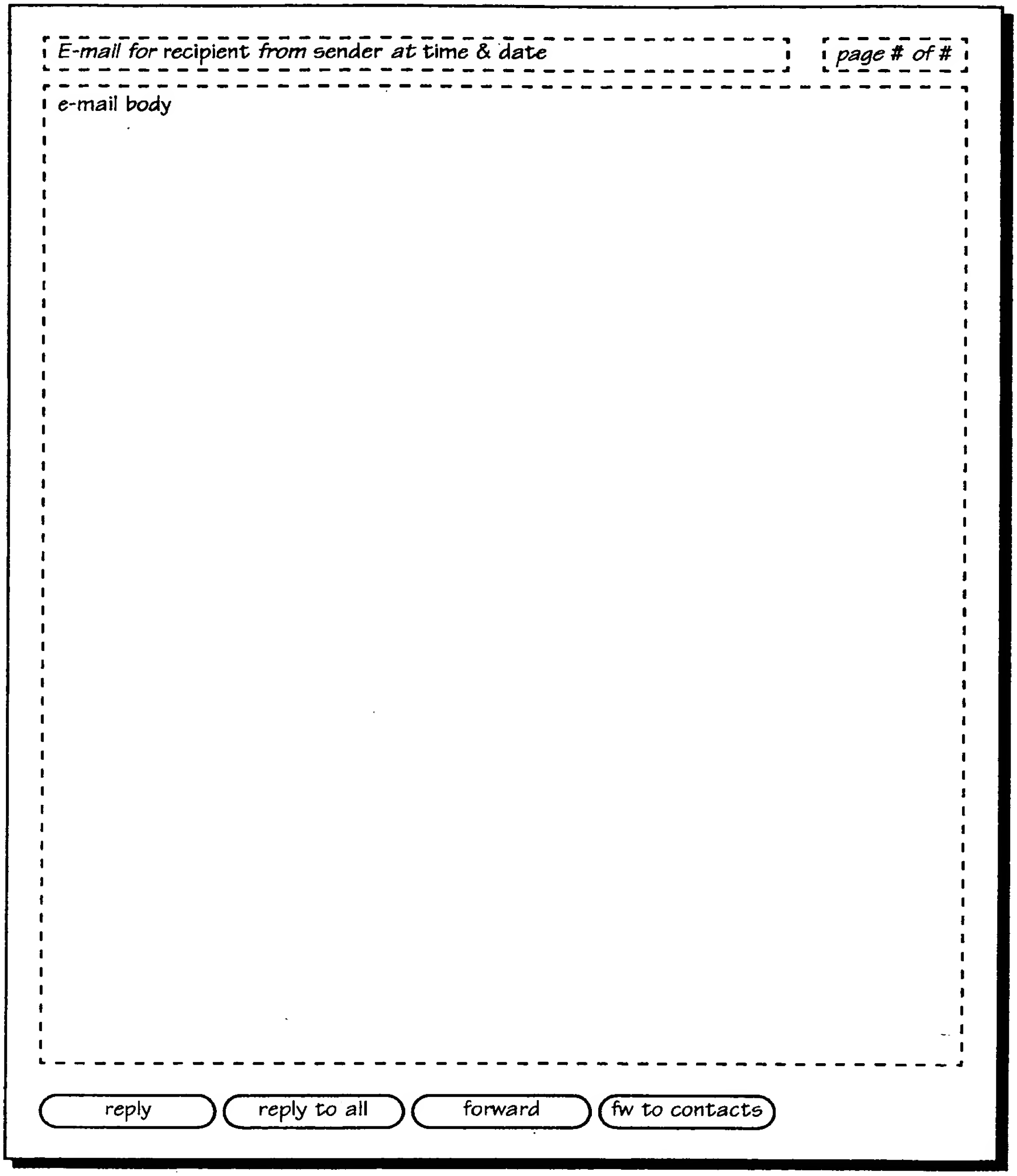


Figure 31. Incoming e-mail second or subsequent page

4.2.3 Edit Contact List

The contact list is edited using the **edit contacts** form (Figure 33). Because the structure of an individual contact group is the same as the structure of the top-level contact list, the same form is used to edit a contact group. The form is obtained by pressing the <edit contact list> button on the help page. It is also printed whenever a contact is added to the contact list, for example when the <add to contacts> button is pressed on an incoming e-mail to add the sender to the contact list.

Each entry on the contact list form shows the name on the contact, and the current e-mail folder in which e-mail from the contact is placed. Each entry has a checkbox which allows the contact to be selected, an <info> button which elicits a page of information about the user (Figure 44), and a <set mailbox> button which allows the e-mail folder associated with the contact to be changed via the **set e-mail folder** form (Figure 35).

The <delete selected> button at the bottom of the form deletes the selected contacts from the contact list (or contact group). The form is reprinted with the deleted contacts removed.

The <copy to group> and <move to group> buttons at the bottom of the form allows the selected contacts to be copied and moved to a particular group via the **copy contacts** form (Figure 34).

The contact list editing user interface flow is illustrated in Figure 32.

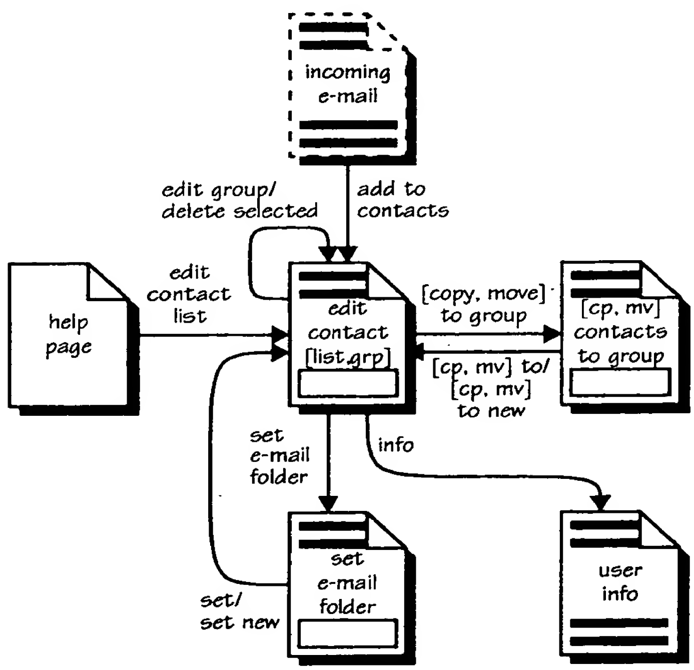


Figure 32. Edit contact [list,group] user interface flow

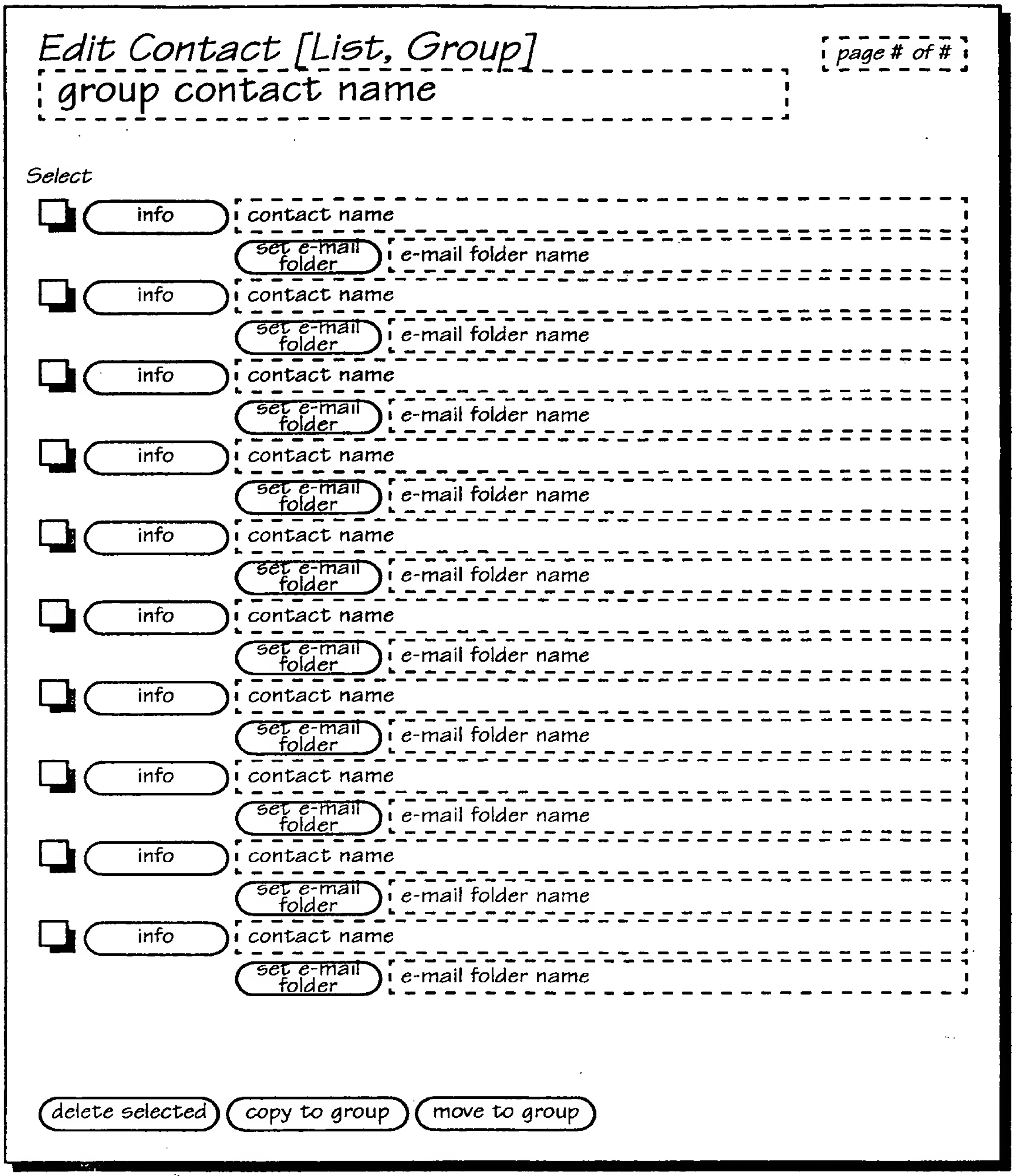


Figure 33. Edit contacts page

The copy (move) contacts form lists all of the groups in the user's contact list, each with a <copy to> (<move to>) button which copies (moves) the selected contacts to the specified group. When a <copy to> (<move to>) button is pressed, an edit contacts form is printed which shows the updated membership of the destination group.

The copy (move) contacts form provides a <new group name> field with an associated <copy to new> (<move to new>) button. This allows a new group to be simultaneously created and selected as the destination for the copy or move.

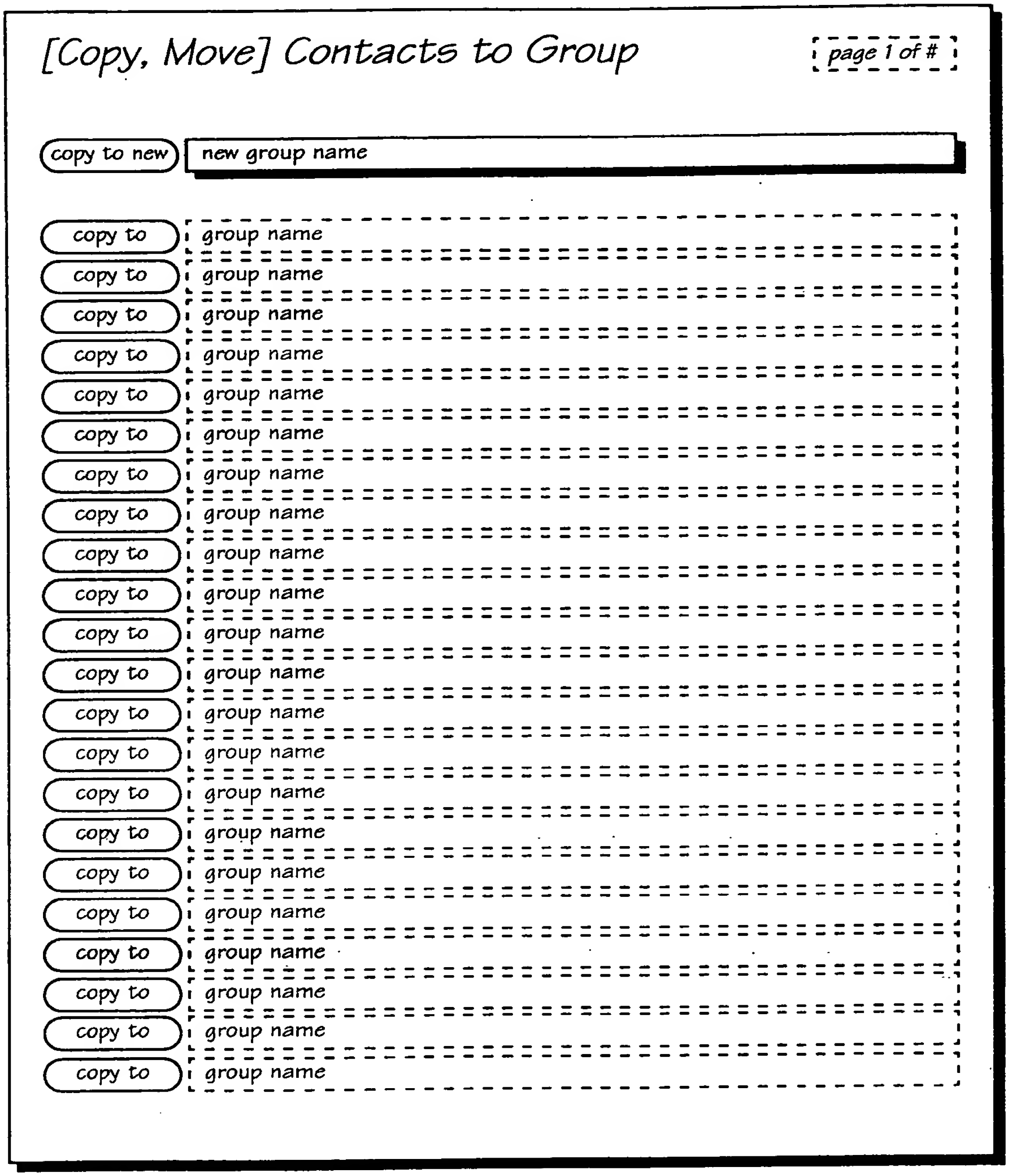


Figure 34. Copy contacts first page

The set e-mail folder form lists all of the folders in the user's mailbox, each with a <set> button which sets the folder as the contact's e-mail folder. When the <set> button is pressed, the edit contacts form is re-printed to reflect the new folder associated with the contact.

The set e-mail folder form provides a <new e-mail folder name> field with an associated <set new> button. This allows a new folder to be simultaneously created and selected as the contact's e-mail folder. The new folder has an associated set of checkboxes which allow the folder's e-mail delivery options to be specified.

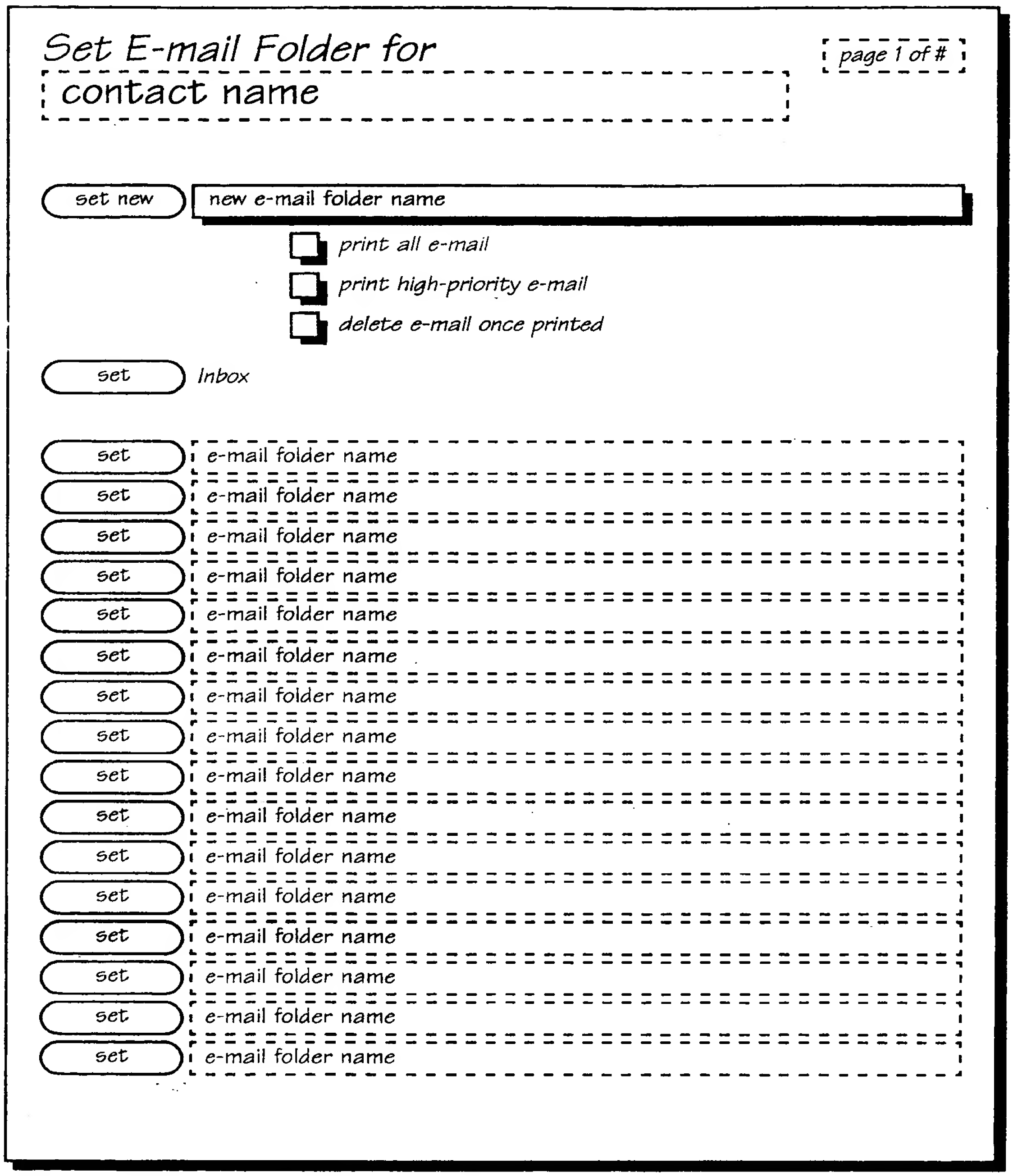


Figure 35. Set e-mail folder first page

4.2.4 List Mailbox

The contents of the user's mailbox are listed using the **mailbox** form (Figure 37) and the **e-mail folder** form (Figure 38). The mailbox form lists all of the user's e-mail folders, each with a set of checkboxes which reflect the folder's e-mail delivery options and allow the delivery options to be changed. Each folder entry also has a checkbox which allows it to be selected. The <count> next to each folder name indicates the number of un-printed (i.e. un-read) e-mails in the folder.

The <delete selected> button at the bottom of the form deletes all selected folders from the mailbox. The mailbox form is re-printed with the deleted folders removed.

The <update options> button at the bottom of the form applies any changes made via the checkboxes to the folders' delivery options.

Each folder entry also has <list> button which lists the contents of the folder via the e-mail folder form (Figure 38).

The mailbox listing user interface flow is illustrated in Figure 36.

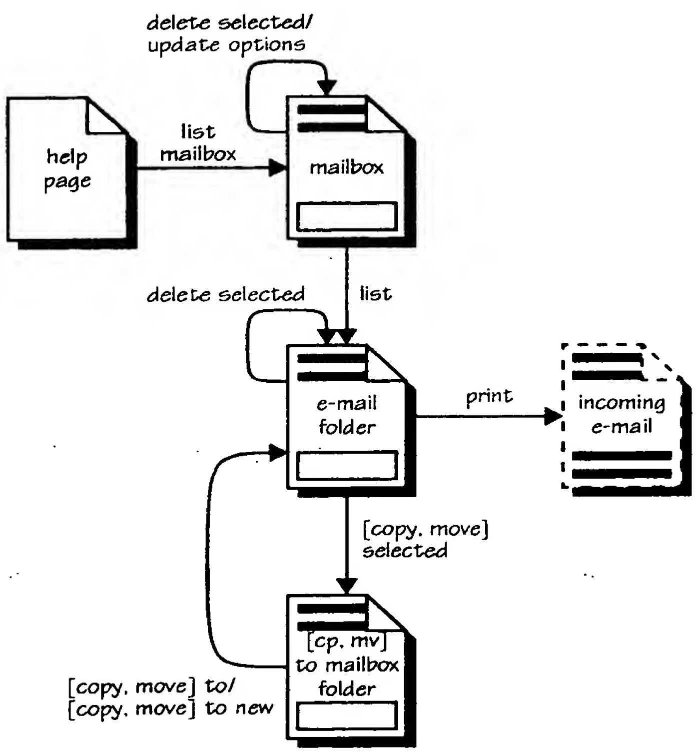


Figure 36. List mailbox user interface flow

The e-mail folder form lists all of the e-mails in the folder. Each single-line e-mail entry shows the sender of the e-mail, the subject of the e-mail, and the date and time the e-mail was received. The list can be sorted by sender, subject, or date and time. Each column which is not the current sort column has a <sort> button which allows the list to be sorted by that column. When a <sort> button is pressed the e-mail folder is sorted by the selected column and the form is re-printed.

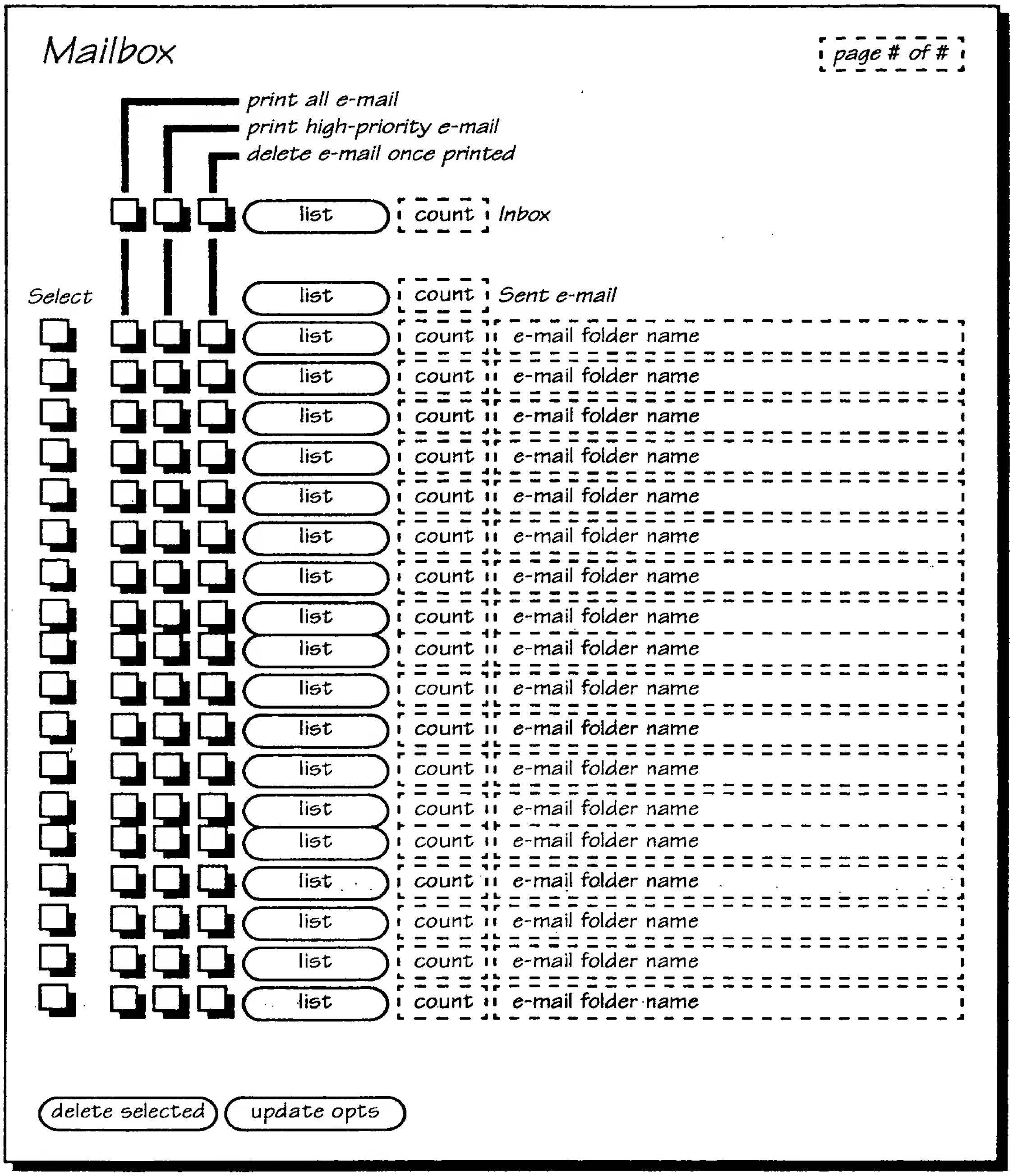


Figure 37. Mailbox page

Where an e-mail is a reply to another e-mail, the subject of the replying e-mail is indented relative to the subject of the e-mail to which it is replying. Thus each thread of a discussion is clearly visible. Entries for e-mails which have never been printed are printed in bold to draw the user's attention.

Each e-mail entry has a checkbox which allows it to be selected, as well as a print button which allows the entire e-mail to be printed in the form of an Incoming E-mail (Figures 30 and 31).

The <delete selected> button at the bottom of the form deletes the currently selected e-mails. The e-mail folder form is reprinted with the deleted e-mails removed.

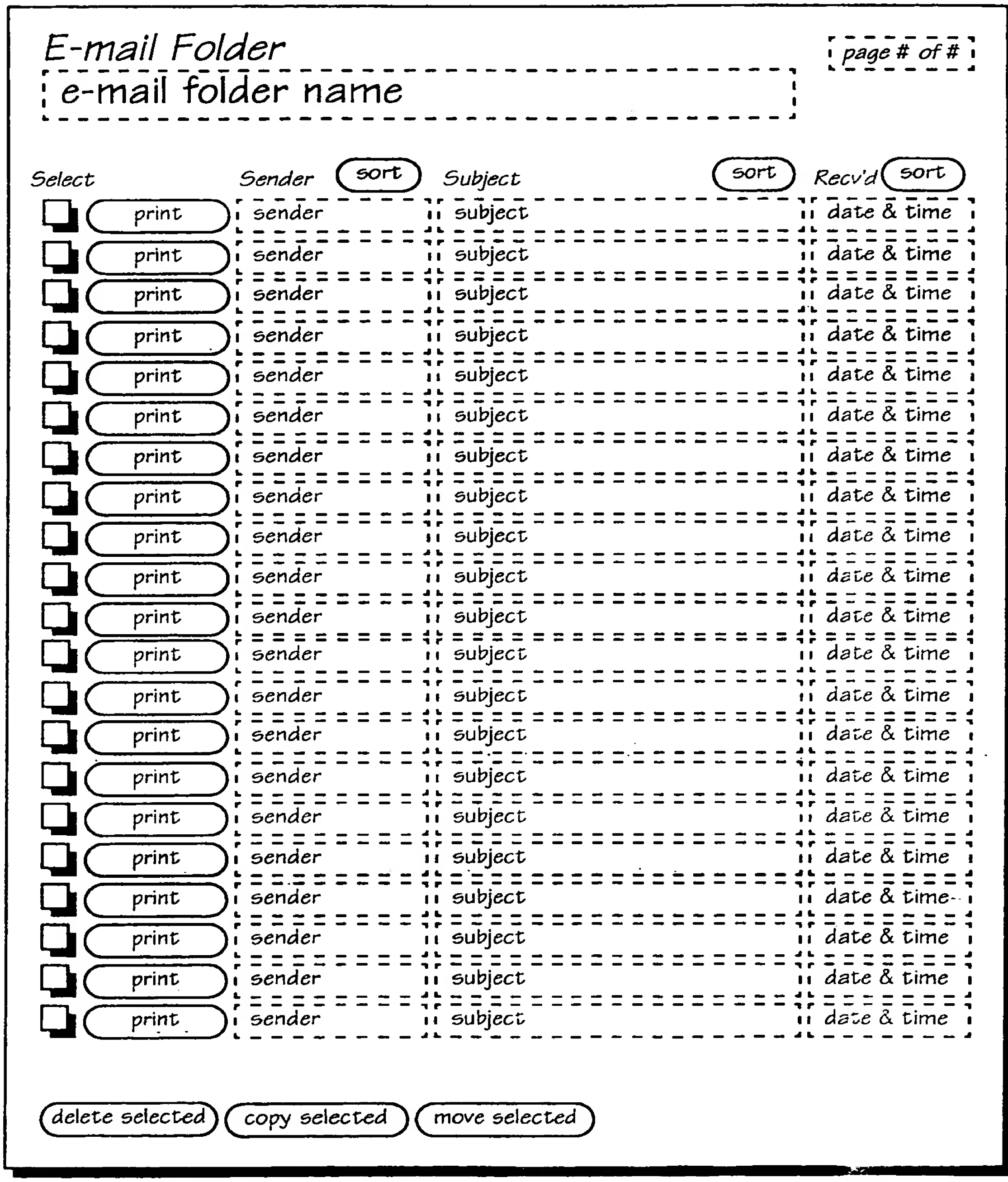


Figure 38. E-mail folder page

The <copy selected> and <move selected> buttons at the bottom of the e-mail folder form allow the currently selected e-mails to be copied or moved to a different e-mail folder via the **copy e-mail** form (Figure 39).

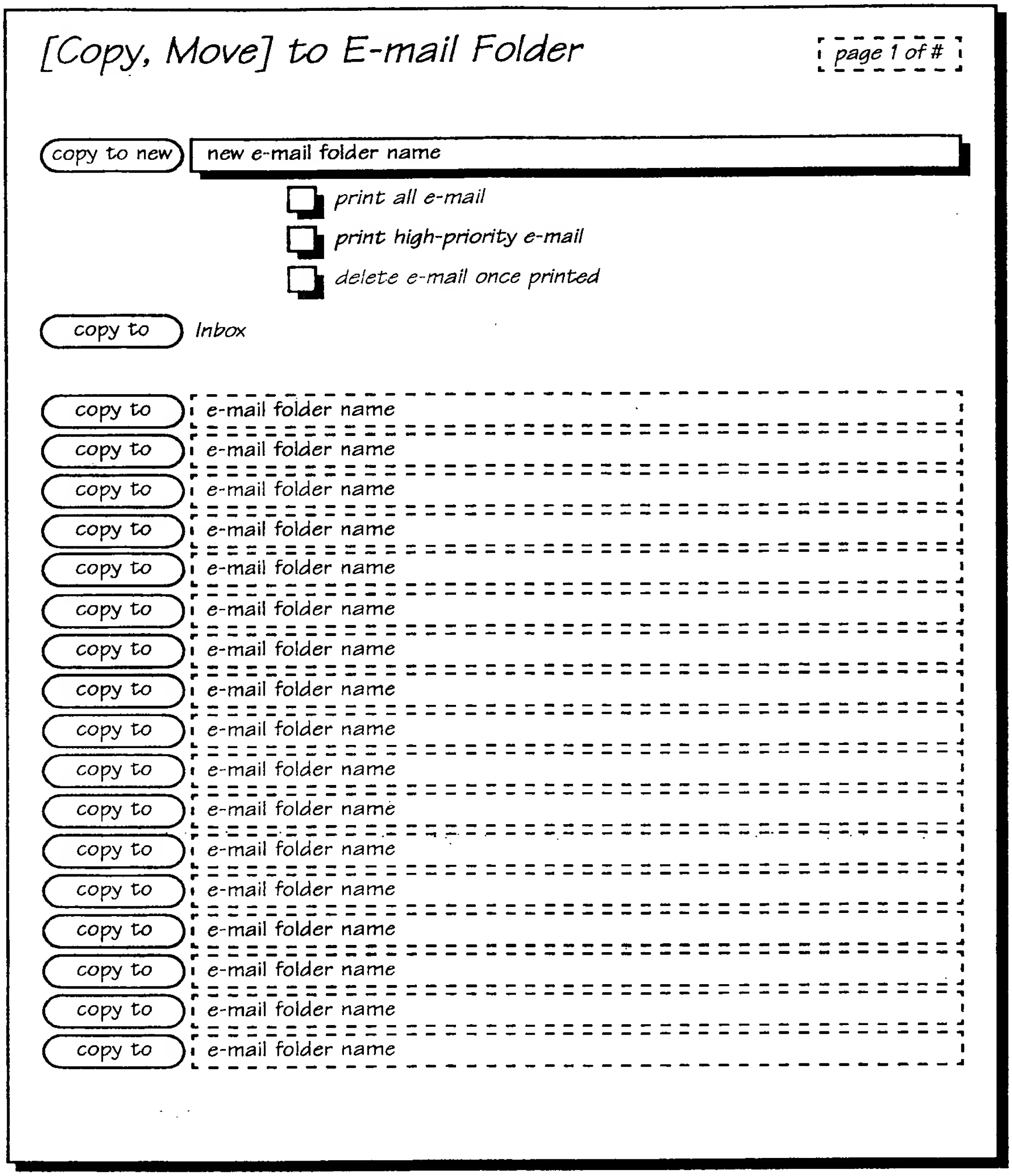


Figure 39. Copy e-mail first page

The copy (move) e-mail form lists all of the folders in the user's mailbox, each with a <copy to> (<move to>) button which copies (moves) the selected e-mails to the specified folder. When a <copy to> (<move to>) button is pressed, an e-mail folder form is printed which shows the updated contents of the destination folder.

The copy (move) e-mail form provides a <new e-mail folder name> field with an associated <copy to new> (<move to new>) button. This allows a new folder to be simultaneously created and selected as the destination for the copy or move. The new folder has an associated set of checkboxes which allow the folder's e-mail delivery options to be specified.

4.2.5 Edit Barred User List

The barred user list is edited via the **edit barred users** form (Figure 41). The form is obtained by pressing the <edit barred user list> button on the help page. It is also printed whenever a user is barred, for example when the <bar> button is pressed on an incoming e-mail to add the sender to the barred user list.

The form lists all of the users in the user's barred user list. Each has a checkbox which allows it to be selected.

The <unbar selected> button at the bottom of the form deletes the selected users from the barred user list. The form is re-printed with the deleted users removed. When a user is deleted from the list that user is no longer barred.

The barred user editing user interface flow is illustrated in Figure 40.

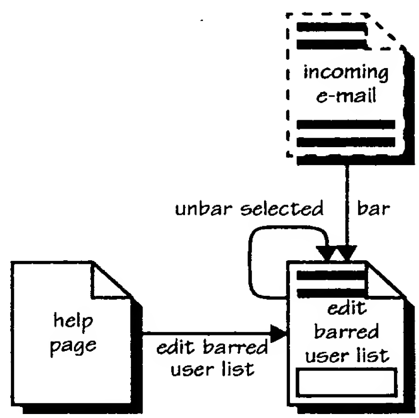


Figure 40. Edit barred user list user interface flow

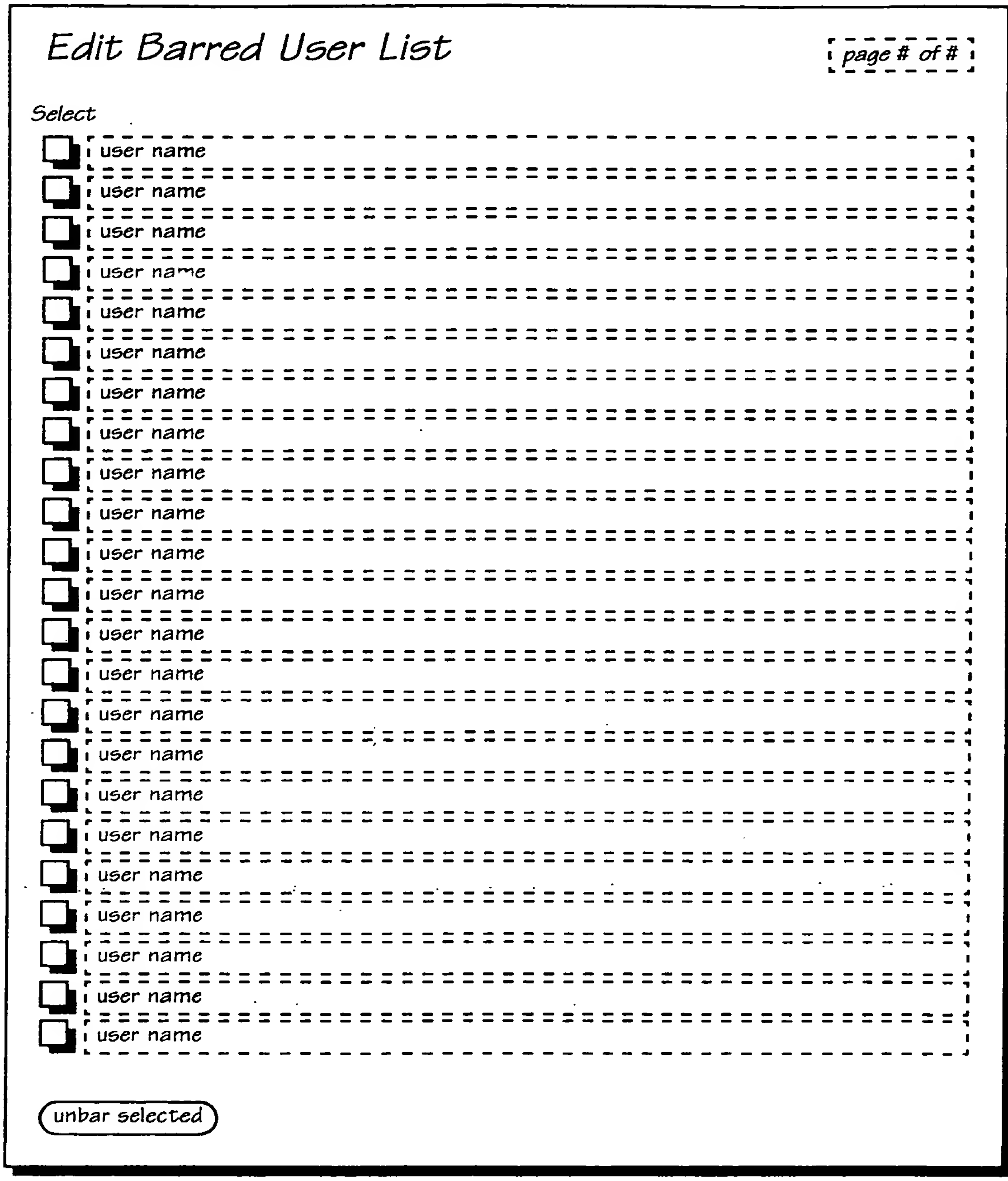


Figure 41. Edit barred users page

4.2.6 Global User Directory

The global user directory, since it is large, is navigated via a directory index (as described in Section 2 on page 6). It is obtained by pressing the <global user directory> button on the help page.

The **global user directory** page (Figure 43) contains an alphabetical list, by family name, of users in the name range covered by the page. Each entry has three associated buttons. The <info> button produces a **user information** page (Figure 44), the <e-mail> button generates an outgoing e-mail form (Figures 27 and 28) addressed to the corresponding user, and the <add to contacts> button adds the user to the user's contact list and prints the updated contact list (Figure 33).

The global user directory page also contains the standard index navigation buttons.

The global user directory user interface flow is illustrated in Figure 42. The index user interface flow and associated page layouts are as described in Section 2 on page 6.

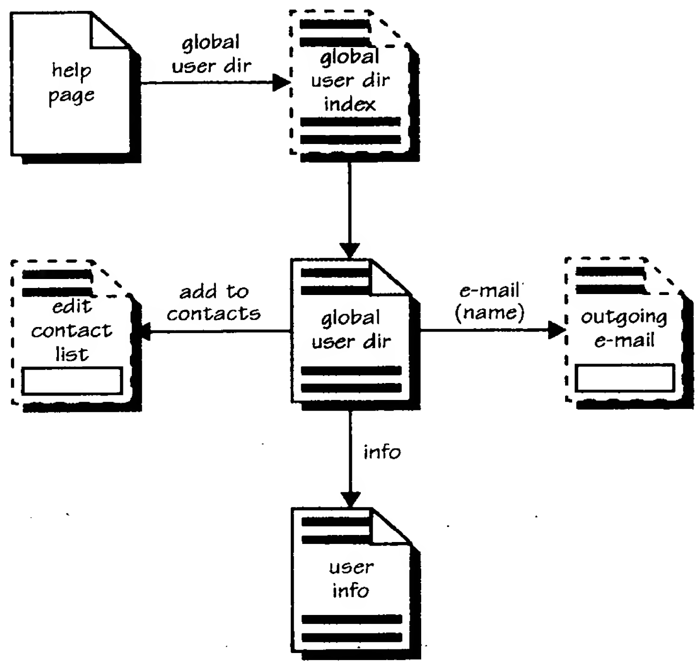


Figure 42. Global user directory user interface flow

The user information page (Figure 44) contains the user's contact details as provided during user registration, subject to the user's privacy preferences. It also contains <e-mail> and <add to contacts> buttons which work in the usual way.

The <Internet e-mail address> is only included if the user information page is printed via the contact list for an Internet contact (as opposed to a Netpage contact).

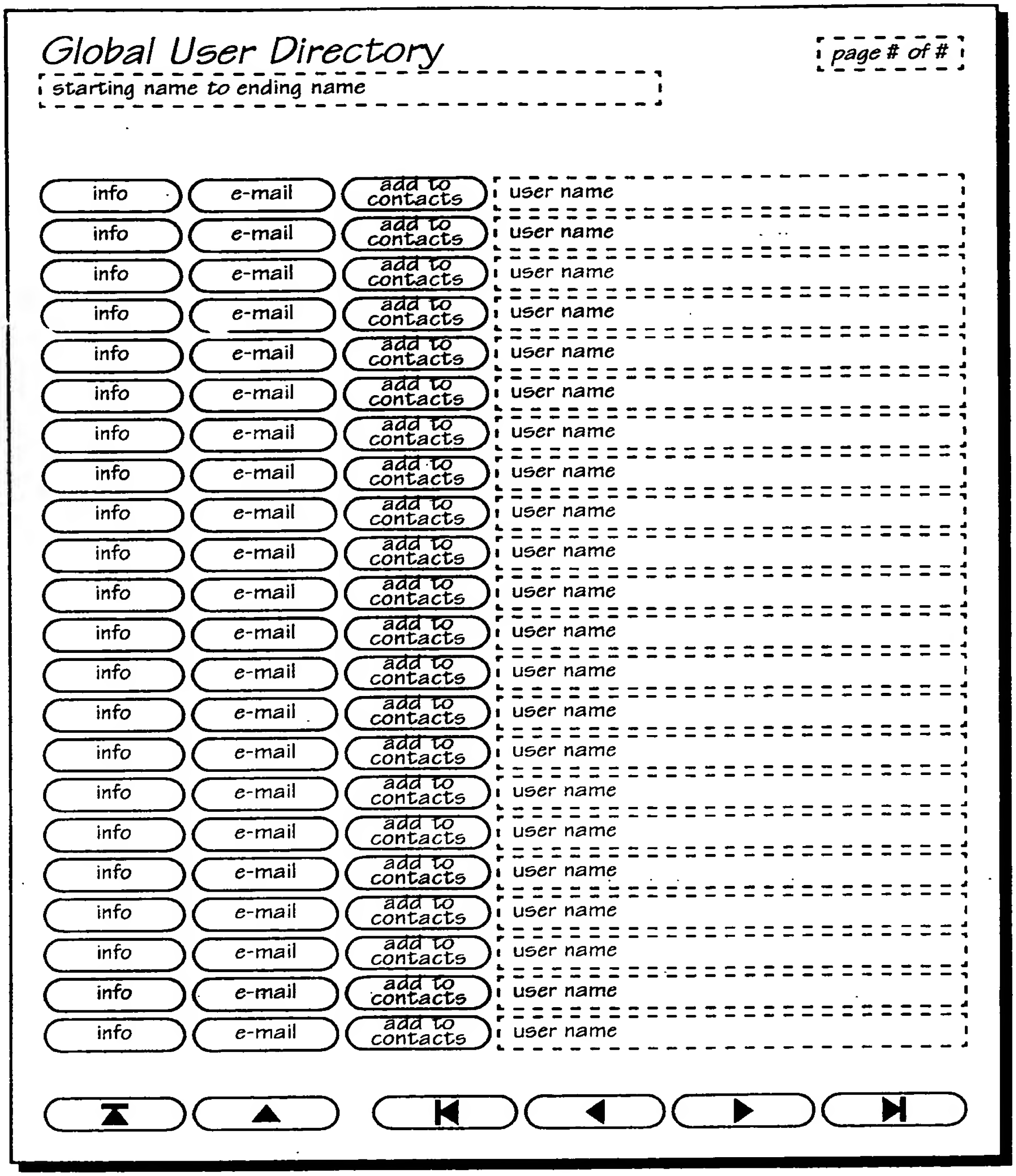


Figure 43. Global user directory page

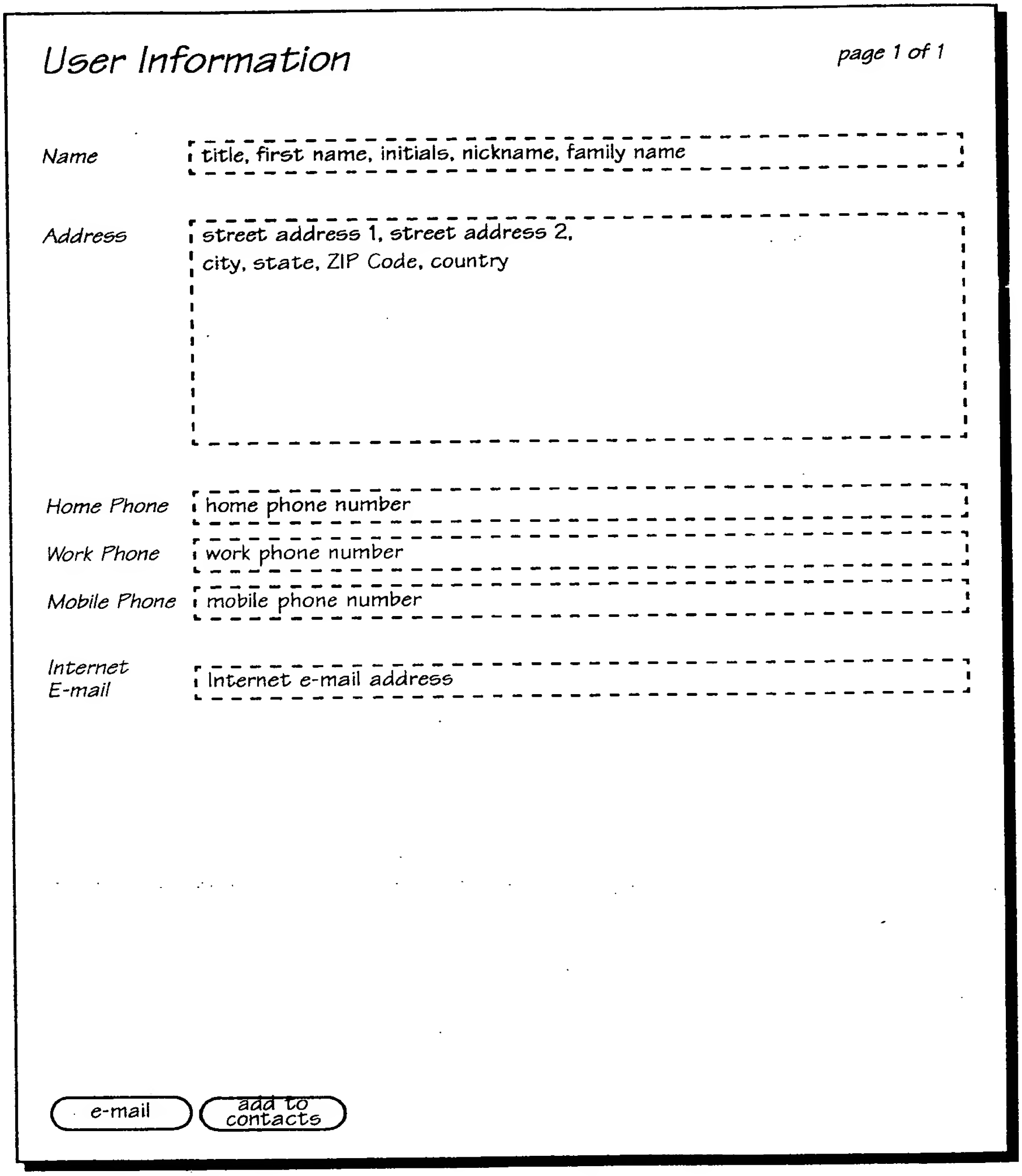


Figure 44. User information page

4.2.7 Business Card

A Netpage business card (Figure 46) serves a dual purpose. While it conveys the user's contact details in the usual way, it also acts as an e-mail authorization token. When the recipient of the card presses the <add to contacts> button on the card, the owner of the card is added to the recipient's contact list in the usual way, but the recipient is also added to the owner's contact list, authorizing the recipient to send e-mail to the card owner, should the card owner only accept e-mail from known contacts. The token only allows the recipient to be added to the owner's contact list once.

The card also contains an <e-mail> button which generates an outgoing e-mail form addressed to the card owner in the usual way.

The business card user interface flow is illustrated in Figure 45.

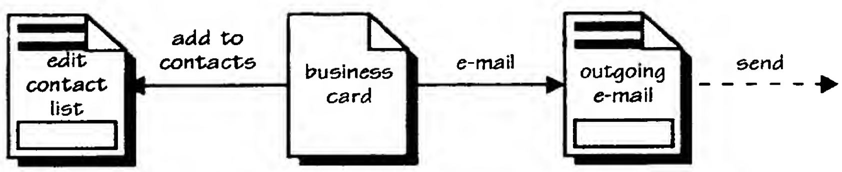


Figure 45. Business card user interface flow

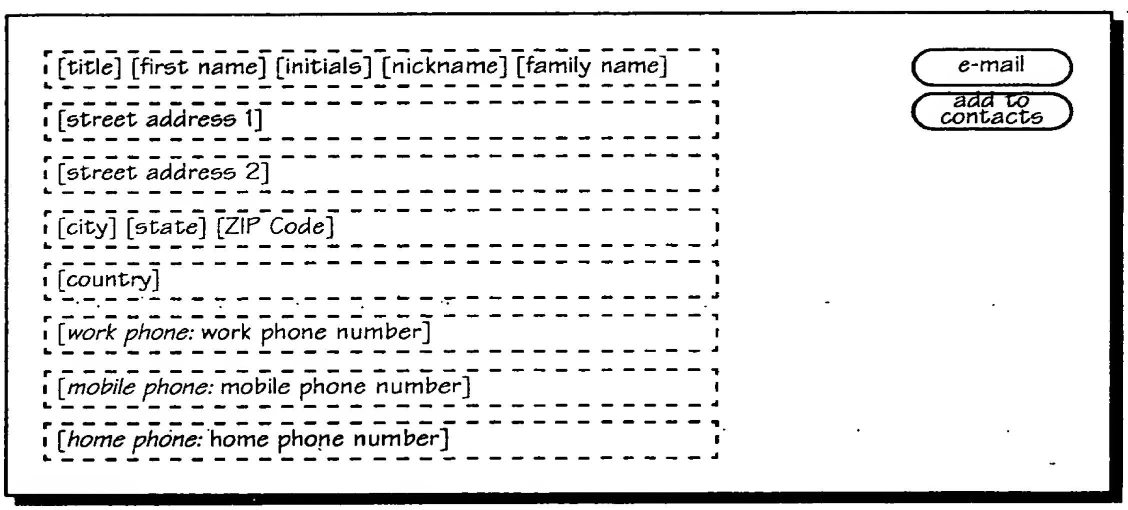


Figure 46. Business card

Since normal Netpage Printers don't support business card paper stock, Netpage business cards are provided by a service bureau on request. The **business card request** form (Figure 48) allows the user to specify which contact details to include on the business card, an optional background texture, and the number of cards required. Once the request is submitted, the cards are automatically printed by the bureau, delivered by mail to the user's address, and the small cost charged to the user's Netpage account.

The business card request user interface flow is illustrated in Figure 47.

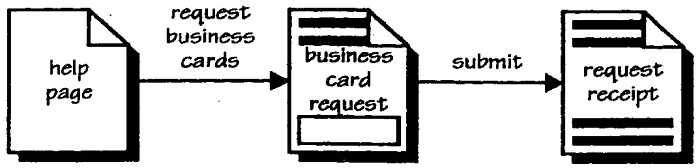


Figure 47. Business card request user interface flow

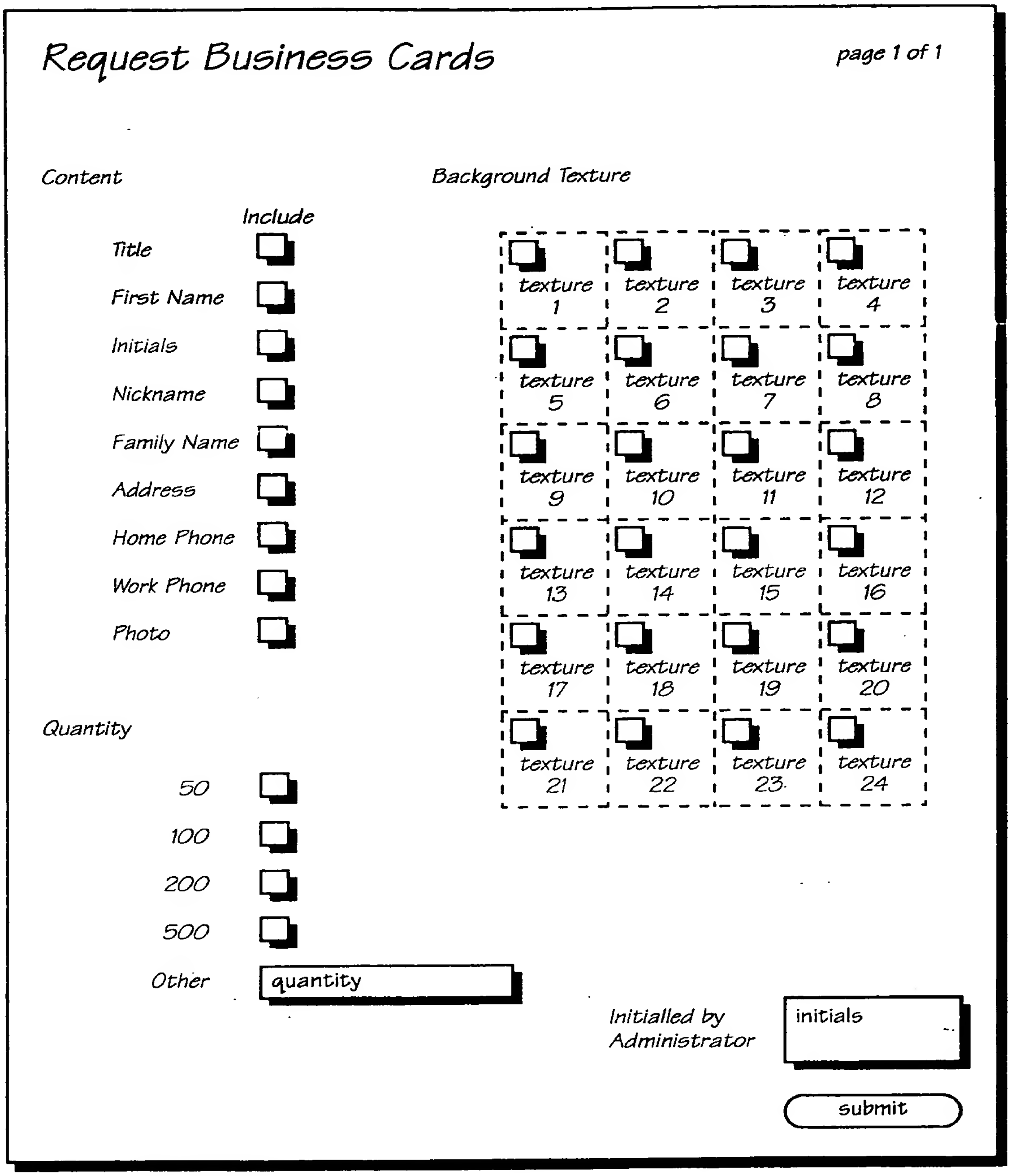


Figure 48. Business card request page

4.2.8 Add Internet Contact

An Internet e-mail user can be added to a user's contact list using the **Internet contact registration** form (Figure 50). Since an Internet user is not otherwise known to the Netpage System, the form allows full name details to be specified, in addition to the Internet user's e-mail address.

The Internet contact registration user interface flow is illustrated in Figure 49.

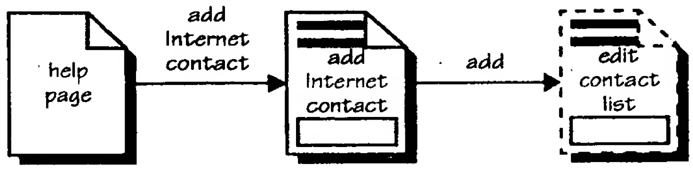


Figure 49. Internet contact registration user interface flow

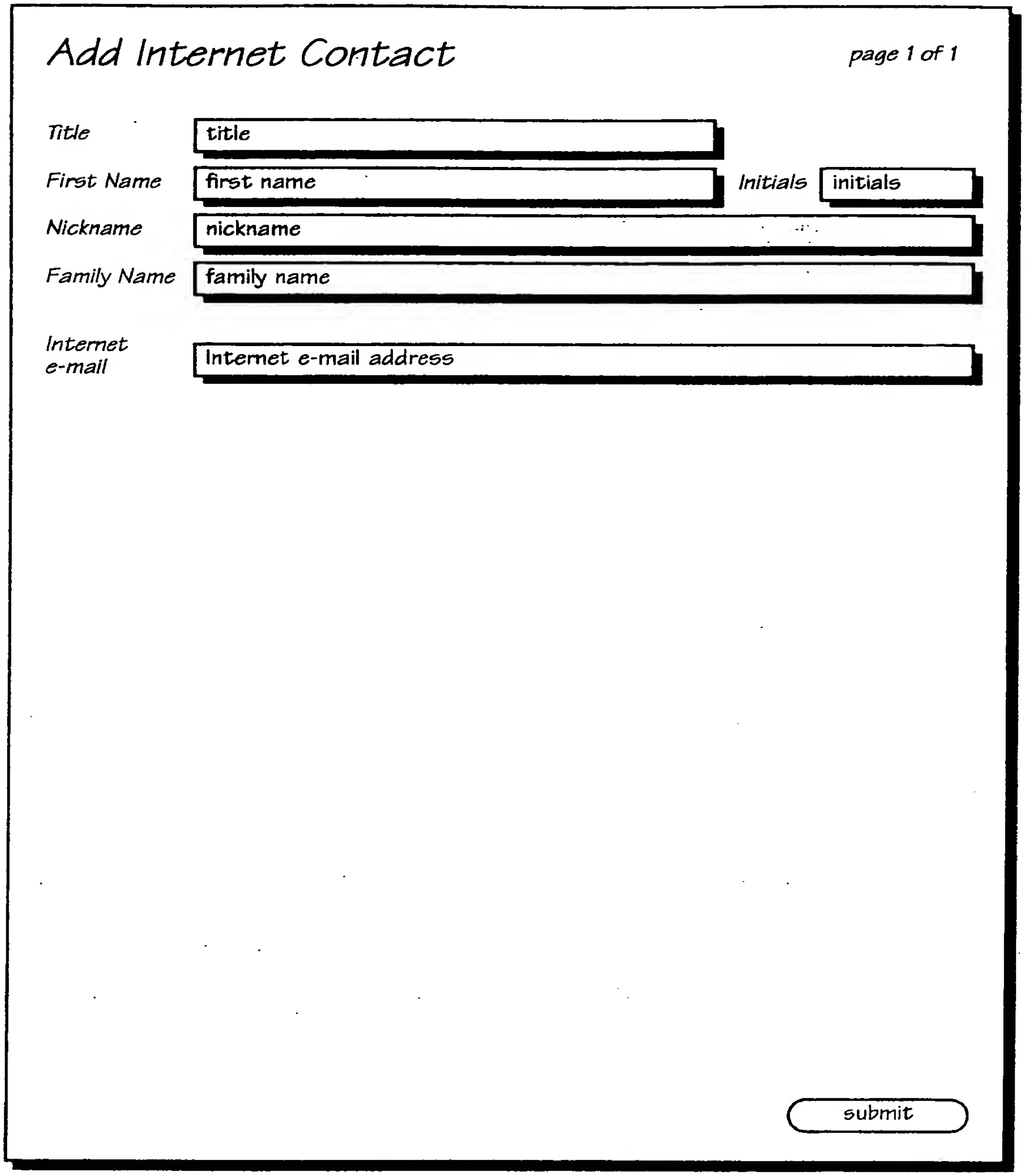


Figure 50. Internet contact registration page

5 Purchasing

The Netpage System provides efficient mechanisms for merchants to offer goods and services to customers, and for customers to select and pay for those goods and services.

A customer can order directly from a catalog browsed and printed dynamically via Netpage, as well as from a pre-printed catalog delivered by traditional means.

The customer's dynamic signature, in combination with the customer's uniquely identified pen, authenticates the purchasing transaction in a way which is impossible to forge. If the customer chooses to use the Secure Electronic Transaction (SET) payment mechanism built into the Netpage System, then the customer's payment card details are never revealed to the merchant, ensuring that card details can neither be accidentally nor maliciously compromised.

The purchasing mechanisms described in this section are generic, and apply to the purchasing of myriad kinds of goods and services.

5.1 Purchasing Object Model

The purchasing object model revolves around merchants and customers.

Each merchant has a globally unique identifier, as well as a name and other details. Each customer is related to a particular merchant, and has an identifier which is unique within the scope of the particular merchant. Each customer represents a particular Netpage user to the merchant. A Netpage user, on the other hand, can be the customer of any number of merchants.

Each merchant offers a range of products for sale, typically presented in the form of an organized catalog. Any Netpage user can browse the merchant's catalog, but the user must become an explicit customer of the merchant before being able to complete an order with the merchant.

The merchant class diagram is shown in Figure 51, while the customer class diagram is shown in Figure 52.

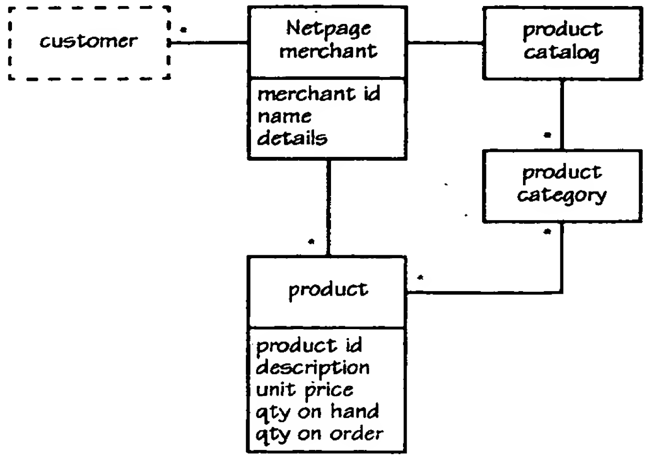


Figure 51. Merchant class diagram

The name and description of a customer derives from the details of the corresponding Netpage user.

Each customer has a history of payment methods and shipping addresses with the merchant. The most recent of each is recorded and used as the default on any new order.

Each customer has an account with the merchant which records the net balance of all invoices and received payments. Account payments, i.e. payments not associated with a particular invoice, are associated directly with the customer account.

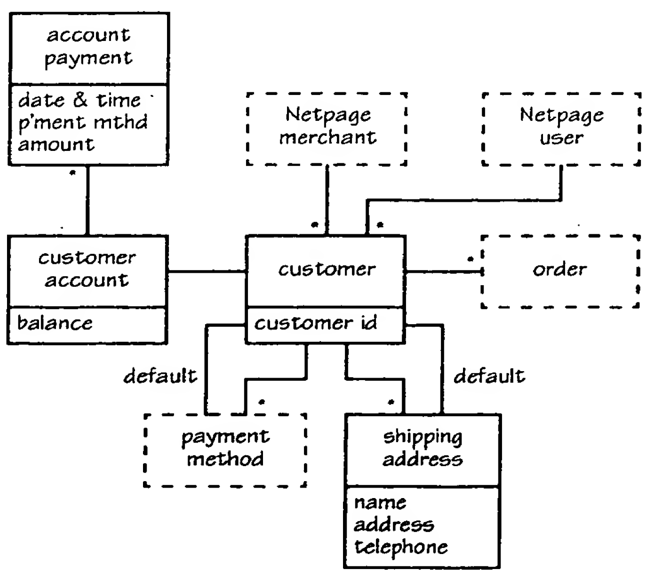


Figure 52. Customer class diagram

Each customer has a history of orders, each in a particular state of completion as reflected by its status. The order class diagram is shown in Figure .

Each order is uniquely identified within the scope of the merchant.

Each order has a status which indicates whether the order is pending (i.e. not yet submitted), active, partially or fully shipped, cancelled or completed. An order is not completed until full payment is received.

Each order consists of a number of order items, each of which specifies the quantity of a particular product offered by the merchant. Each order item also includes a backorder quantity and a quantity shipped. The backorder quantity is used to register a claim on future stock when there is insufficient current stock to fully satisfy the order item. The quantity shipped is used to record the cumulative quantity actually shipped.

Each order is associated with a particular payment method, shipping address, and shipping method.

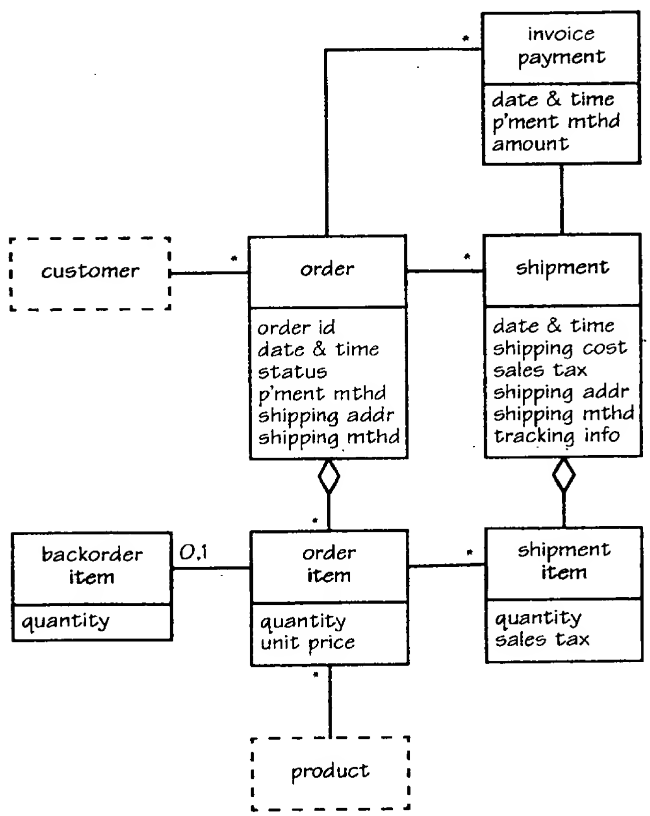


Figure 53. Order class diagram

The most common of the many possible payment methods are shown in the payment method class diagram in Figure 54. The set of payment methods supported by a particular merchant will be specific to that merchant, and may be different for different customers of the merchant depending on their credit ratings etc.

When the Netpage account payment method is used, the user's Netpage account is debited, and the transaction appears on the user's Netpage account statement.

When the customer account payment method is used, the user's customer account is debited, and the user is invoiced for payment.

When the pro forma invoice payment method is used, a pro forma invoice is issued requesting pre-payment of the order. When payment is received, the goods are shipped.

When the cash on delivery payment method is used, payment is expected when the goods are delivered.

When the SET payment card payment method is used, the corresponding SET payment card linked to the user is used to make payment according to the normal SET protocol.

When the payment card payment method is used, the specified payment card is used to make payment according to the normal protocol between the merchant and their acquiring bank or institution which handles the particular payment card type.

Invoice and pro forma invoice payments are normally made with a check or money order. Cash on delivery payments are normally made with a check or cash.

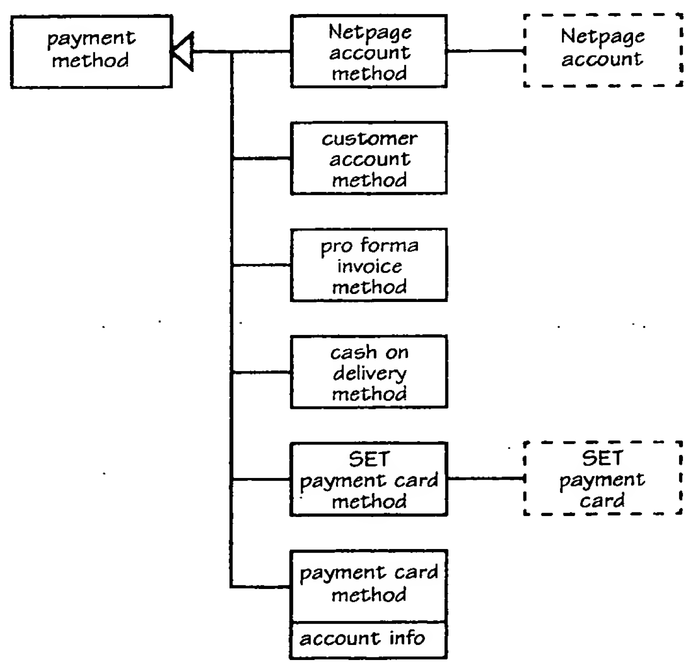


Figure 54. Payment method class diagram

The set of shipping methods which applies to a particular order depends on whether the order's shipping address is domestic or international with respect to the location of the merchant's warehouse. Typical domestic shipping methods include standard shipping, second day air, and next day air. Typical international shipping methods include standard shipping, airmail, and international priority.

The most common shipping methods are shown in the shipping method class diagram in Figure 55.

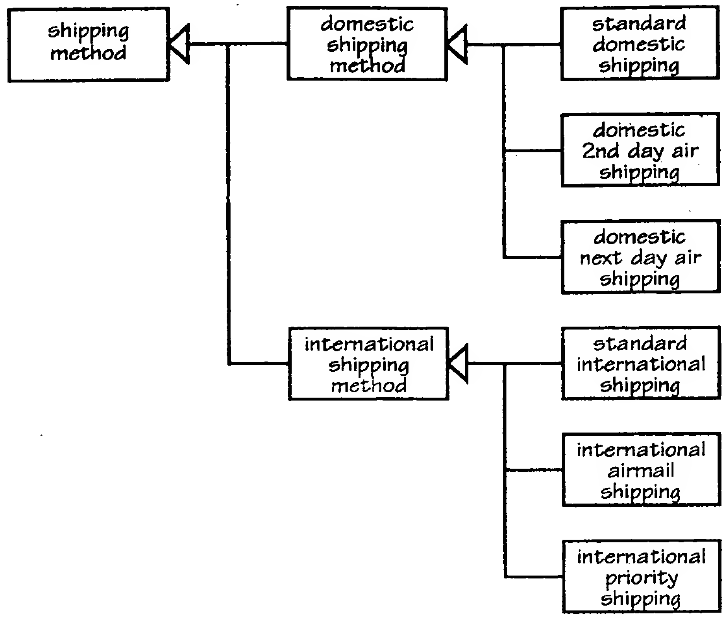


Figure 55. Shipping method class diagram

5.2 PURCHASING USER INTERFACE

The status of the order determines the allowable set of operations on the order. While the order is pending, the user can browse the merchant's catalog and add items to the shopping cart which represents the order. After the order is submitted, the user can still amend the payment method, shipping address and shipping method until the order is fully shipped. A change obviously only applies to items not yet shipped when the change is made. The user can delete order items or change the quantity ordered until the items are actually shipped. Quantity increases are subject to availability and payment approval.

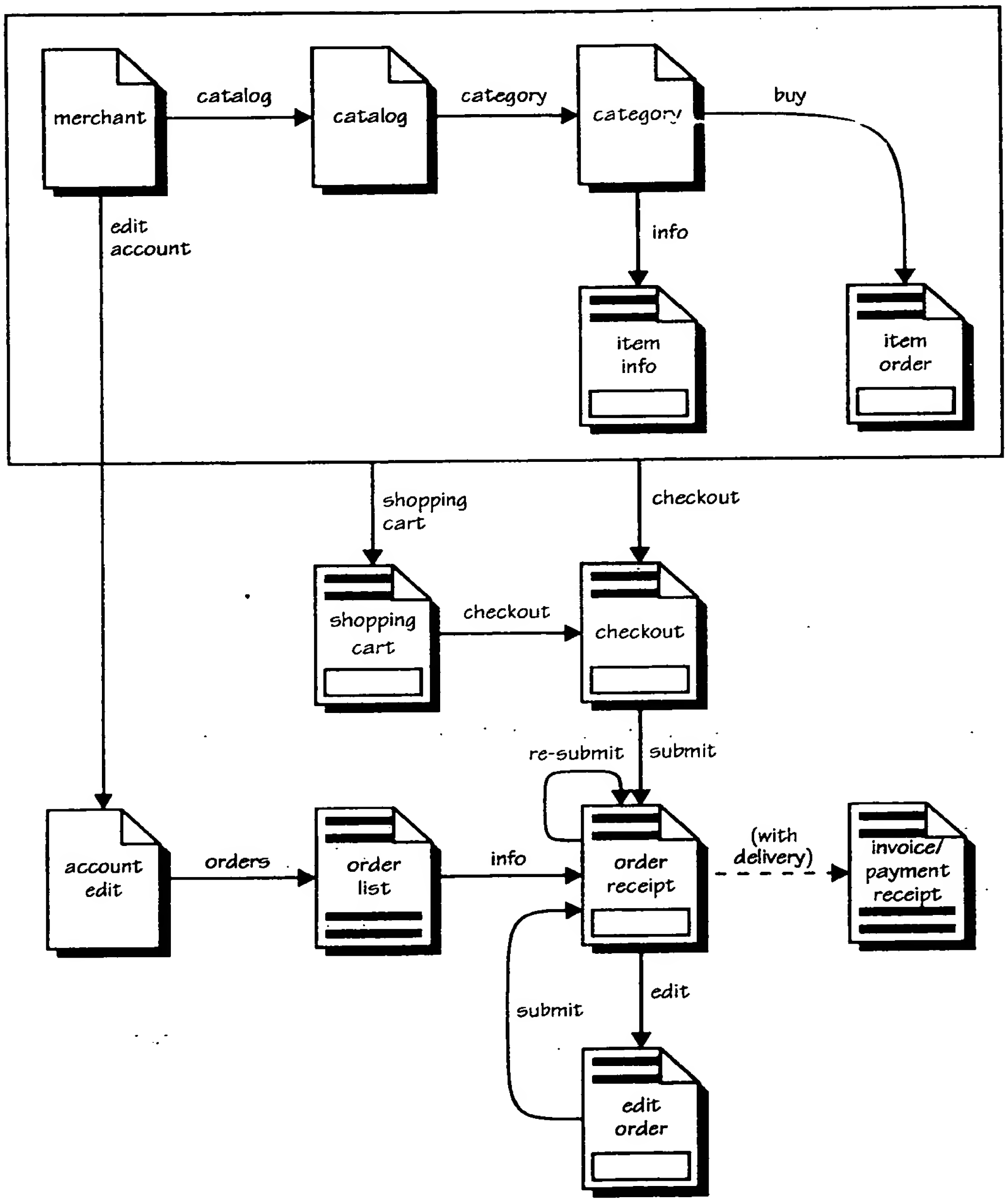


Figure 56. Online purchasing user interface flow

The overall purchasing user interface flow is illustrated in Figure 56.

5.2.1 Locate Merchant

A merchant can be located by browsing or searching the global Netpage directory or the user's own bookmarks. A merchant may also be encountered in the form of an advertisement in a Netpage publication. Both a directory entry and an advertisement typically link to the merchant's main page. This in turn links to the merchant's product catalog. If the user already has a copy of the merchant's main page from a previous encounter, then the catalog is immediately available.

5.2.2 Browse Catalog

A small product catalog is typically organized into a set of categories. The main catalog page simply lists the categories, and each category page lists all of the products in the category. This is indicated in the user interface flow in Figure 56.

If the product catalog is large, then it may still be organized into categories, but it may be impractical to print the entire contents of a category when a user wishes to browse the category. Instead the directory navigation techniques described in Section 2 can be used. A catalog search will often be specific to the kinds of products represented in the catalog.

5.2.3 Order Item

Many applications involve the user maintaining a collection of items. In the screen-based version of the application, the user's collection is typically shown every time an item is added or removed. The user therefore receives immediate feedback when the collection changes. The collection may reflect an e-mail recipient list, a shopping cart, etc.

In the Netpage version of the application, it is less practical to re-print the collection every time it changes. Item deletions and quantity changes can be marked up visibly, and so do not necessarily require immediate re-printing. Item insertions, however, are more problematic. In some cases the entire source list can be used as the input form, allowing it to be marked up visibly. This may be particularly apt if the source list already represents a user-specific subset of the complete source list, e.g. last week's shopping list rather than the grocery store's complete range. There will be cases, however, where the user will primarily be making selections from a complete catalog. The catalog may even be many hundreds of pages long, offset printed, and delivered in the mail. Since it is intended for multiple use, it is not suitable for visible markup.

One way to order via a large catalog is to print intermediate product-specific pages. Each catalog entry has both an <info> and an <buy> button. The <info> button elicits a full-page description of the product with a quantity field which, if filled in, adds the product to the shopping cart *and* provides a visible record of the item order. This avoids having to print the contents of the shopping cart every time an item is added to it. The <buy> button elicits a similar full-page description of the product but automatically adds it to the shopping cart. It contains a quantity field, already filled in with the quantity ordered by default, which can be used to vary the quantity ordered.

Depending on the nature of the order, it may be practical to also list the contents of the shopping cart on the product-specific page. However, it is the fact that the product page gives product-specific information that is of primary value to the user.

The product page is visibly linked to the order itself because it is tagged with the name of the current user, the order transaction number, and the order date and time.

5.2.4 Complete Order

The shopping cart represents the contents of the current order. The shopping cart is accessible from any merchant page, whether the page is part of the current order form or not. The shopping cart provides a convenient place to review the order and make final changes.

The checkout is also accessible from any merchant page. It provides a place to specify shipping and payment information, and finally to confirm the order.

The checkout form is broken up into five sections: shipping address, shipping method, payment method, order items, and order confirmation. Sections have default selections based on the user's most recent behavior.

The order confirmation section summarizes the order and captures the user's signature. It shows the current shipping address, shipping method, payment method, and order total.

Any change the user makes to the order on the checkout form is reflected on the form, since the change is made with the inking pen. It is not reflected, however, in the order summary in the order confirmation section. To produce a consistent checkout form, the user can simply press the <update> button at the bottom of the form after making a number of changes. An updated checkout form is immediately printed.

The <submit> button, when pressed, submits the order. Any handwritten change to any item information page, shopping cart page, or checkout page which is part of the order form is automatically interpreted and applied to the order. The order receipt, printed in response to the order submission, reflects the actual contents of the order.

The checkout form consists of a header (Figure 57), a section for selecting a previously used shipping address or specifying a new address (Figure 58), a section for selecting a shipping method (Figure 59), a section for selecting a previously used payment method or specifying a new payment card (Figure 60), a section for reviewing the list of order items and making last-minute changes (Figure 61), a section for signing the order to confirm payment (Figure 62), and a footer which contains the <update> and <submit> buttons.

The printed checkout form will vary in length depending on the number of previously used shipping addresses and payment cards, and the number of items in the order. In many cases it will fit conveniently on two pages printed on a single sheet of paper.

The order receipt lists the items ordered and summarizes the order parameters in much the same way as the order confirmation section of the checkout form (Figure 62).

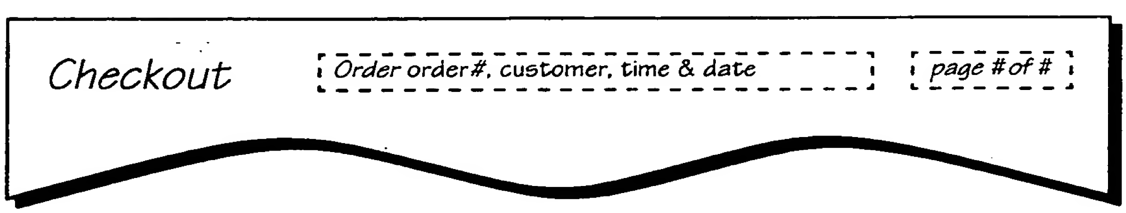


Figure 57. Checkout page header

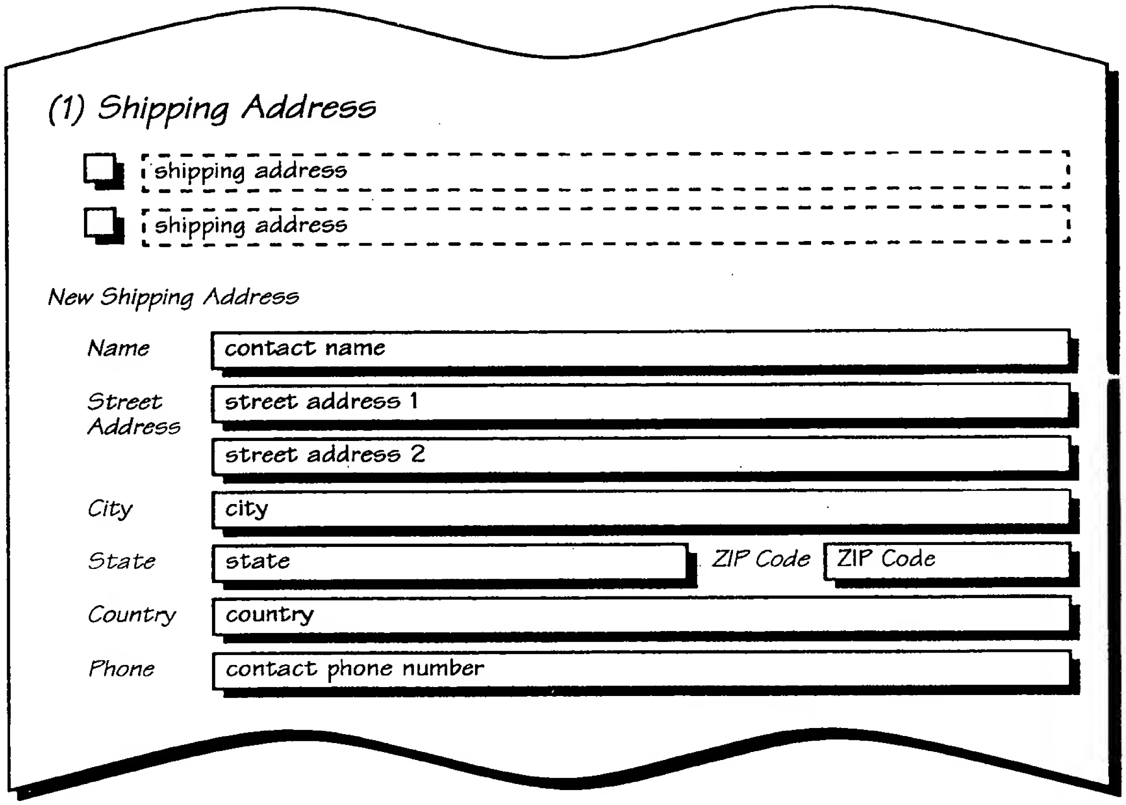


Figure 58. Checkout page shipping address

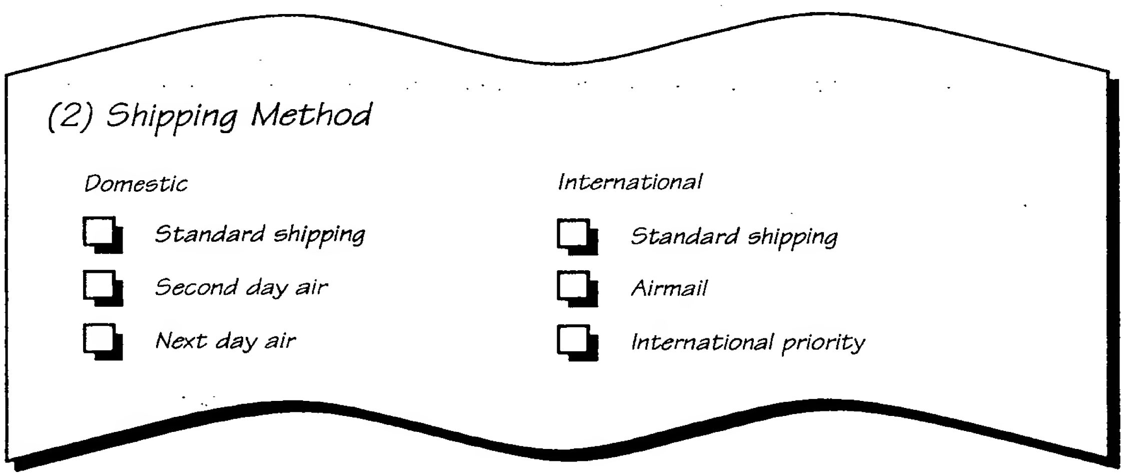


Figure 59. Checkout page shipping method

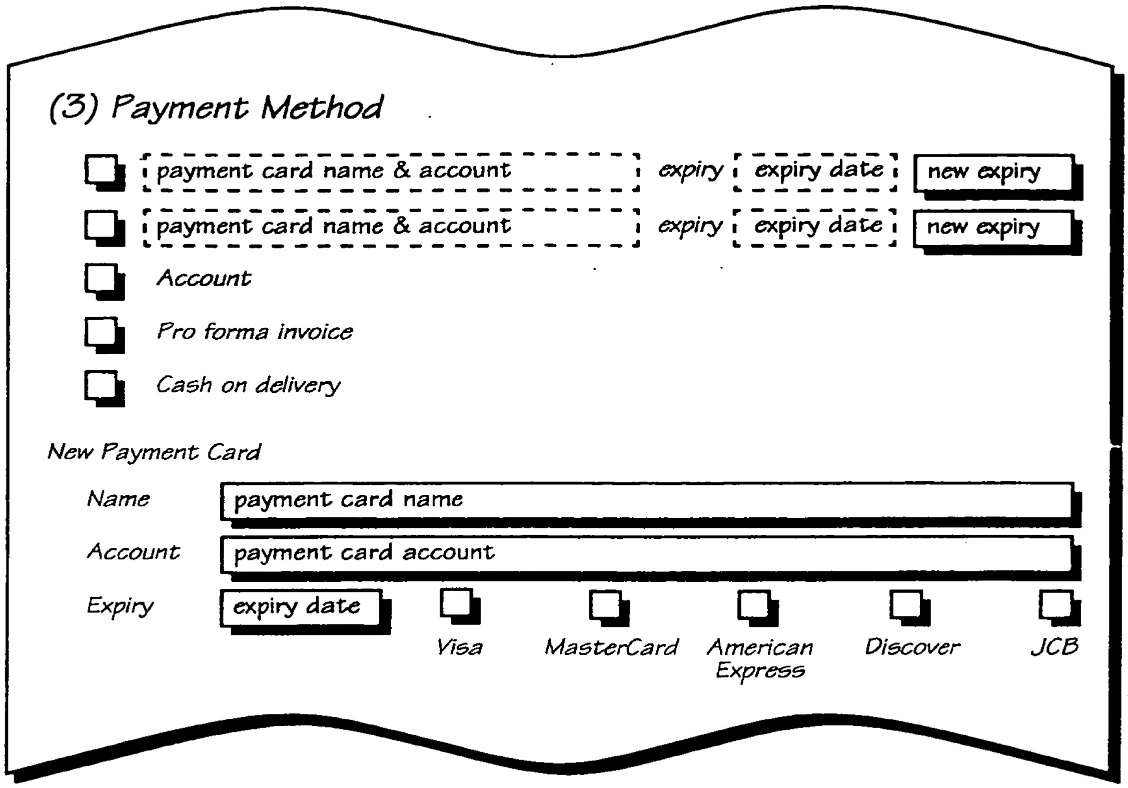


Figure 60. Checkout page payment method

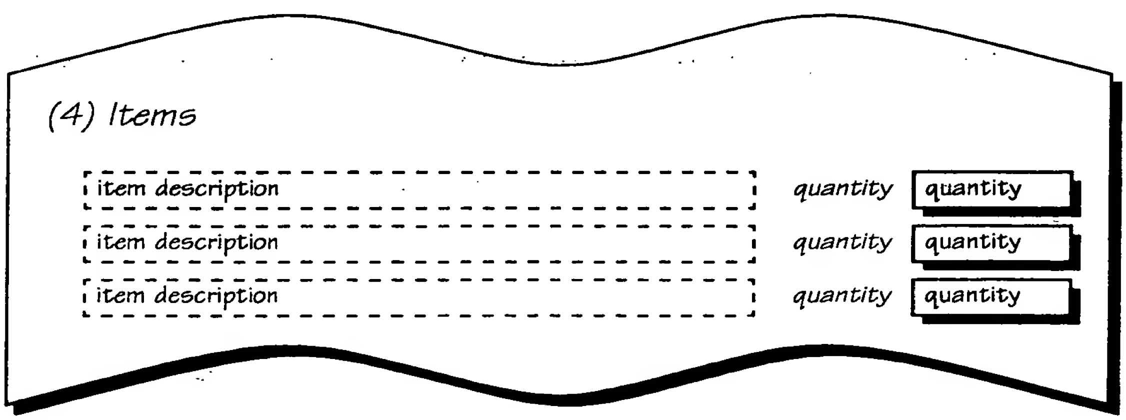


Figure 61. Checkout page item list

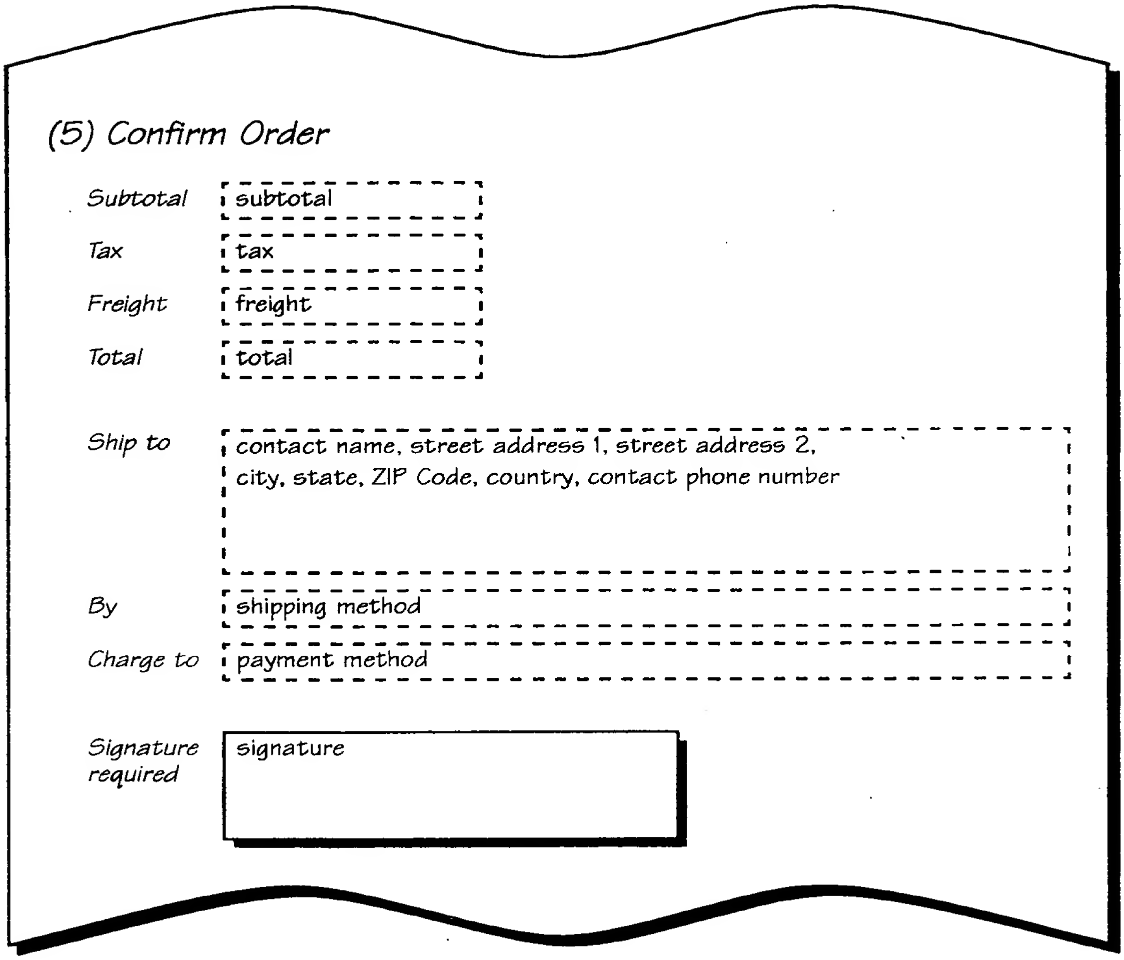


Figure 62. Checkout page order confirmation

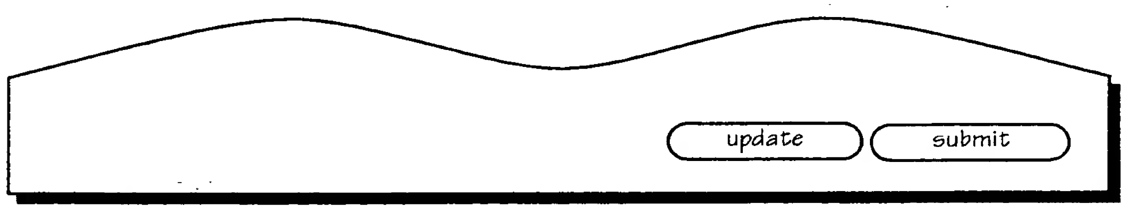


Figure 63. Checkout page footer

5.2.5 Edit Account

Users can edit their customer details maintained by the merchant, including their list of shipping addresses and their list of payment methods. They can also print their order list,

and details of individual orders. They can edit any order which has not been fully shipped, as described earlier.

The order list gives the status for each order. Pressing the <info> button associated with each order yields a copy of the order receipt, but with an updated status for each order item. The status of an order item derives from the status of the order it is part, as well as the presence of shipment items and backorder items associated with the order item.

5.2.6 Edit Order

The order receipt allows the quantities of individual order items which are still active to be altered. Once alterations are made, the user must press the <submit> button. An updated receipt is produced.

To edit order parameters, the user can press the <edit> button on the order receipt. This produces an order editing form which is similar in structure to the checkout form. Once alterations are made, the user must press the <submit> button. An updated receipt is produced

5.2.7 Receive Shipment

The shipment is accompanied by payment receipt, or, if the payment method is "cash on delivery", an invoice which allows the user to pay the delivery person by traditional means (e.g. cash, check, or payment card not processed through Netpage), or using a Netpage payment method listed on the form, simply by marking the appropriate payment method and signing the form with a Netpage pen.

REFERENCES

6 Silverbrook References

7 Other References

[1] Booch, G., J. Rumbaugh and I. Jacobsen, *The Unified Modeling Language User Guide*, Addison-Wesley 1999

[2] Rumbaugh, J., I. Jacobsen and G. Booch, *The Unified Modeling Language Reference Manual*, Addison-Wesley 1999

Appendix 3
(Pages 88)

Netpage Printer
Design Description

draft version 0.3, 7 September 1999

Silverbrook Research Pty Ltd
393 Darling Street, Balmain
NSW 2041 Australia
Phone: +61 2 9818 6633
Fax: +61 2 9818 6711
Email: info@silverbrook.com.au

Confidential

Document History

Version	[illegible]	[illegible]	[illegible]
0.3d	7 September 1999	Paul Lapstun Toby King Simon Walmsley Garry Jackson Kia Silverbrook	Split off from Netpage System Design Description. Work in progress...
0.2d	1 June 1999	Paul Lapstun Toby King Simon Walmsley Garry Jackson Kia Silverbrook	Initial draft issue.

Contents

PRINTER PRODUCTS

1 Overview

Netpage Printers are intended for use in domestic, commercial, corporate and hospitality environments. They all utilize pagewidth Memjet printheads [4], and are based on a simple and reliable straight paper path, passing through a Memjet transfer roller printhead mechanism. In most cases the printed page is glued along one edge and adhered to the previous page to form a final bound document that can be 1 page or 500 pages thick. They all interact with the wireless Netpage Pen.

Netpage Printers come in various forms: wall-mounting, tabletop, portable, and pocket versions.

2 Wallprinter

2.1 WALLPRINTER

A low-cost, wall-mounted, base model with a duplex 8½" Memjet printhead array that accepts a full ream of US Letter paper in a vertical format as shown in Figures 1, 3 and 4. Paper is placed into a hinged top tray down onto a sprung platen and registered under edge guides before being closed. Figure 2 shows the access to the paper and ink cartridge.

A replaceable cartridge containing cyan, magenta, yellow, and infrared inks and glue is also accessible when the tray is open. It connects via a series of self-sealing connectors to hoses that transmit ink and glue to their separate locations. The cartridge consists of a thin wall drawn aluminum casing that accommodates four ink bladders and a single glue bladder into an injection molded connector base. This is a fully recyclable product with a capacity for printing and gluing 3000 pages (1500 sheets). It is protected from forgeries by use of an authentication chip [1,2].

When closed, a release mechanism allows the platen to push the paper against the pick-up roller assembly, where it is fed directly into the duplex Memjet printhead assembly. From there, the sheet passes a momentary action glue wheel with powered spike wheels, where it has glue applied to the vertical edge as it passes through. The glue wheel is capped when not in use and is operated by a powered camshaft.

The printed sheet is fed down to a binding platen that operates with a closed steel wire loop system of pulleys, runners and a powered axle. This provides the necessary speed to push the sheet forward onto the rear of a previous sheet, glue/bind it and return to the home position to accept the next printed sheet in less than 2 seconds. A motorized paper tapper assembly aligns the sheets in a simultaneous operation.

When a document is bound and finished, a powered exit hatch with a tamper sensor opens. Plastic foils work together with the hatch to direct the finished document to the back of the collection tray and feed any further documents into the tray without hitting existing ones. The collection tray is molded in clear polycarbonate and pulls out of its socket under a certain loading. Access for removing documents is provided on three sides.

The printer has a main PCB that accommodates all major circuitry components including external data jacks. A flex PCB runs from the main PCB to the paper tray and has three different color LEDs and a push button. The LEDs indicate "on", "ink out", "paper out", and "error". The push button is a "help" interface that prints out a simple instruction sheet and a compact features directory for the user. The unit is powered by an internal 110V/220V power supply that is connected before wall-mounting.

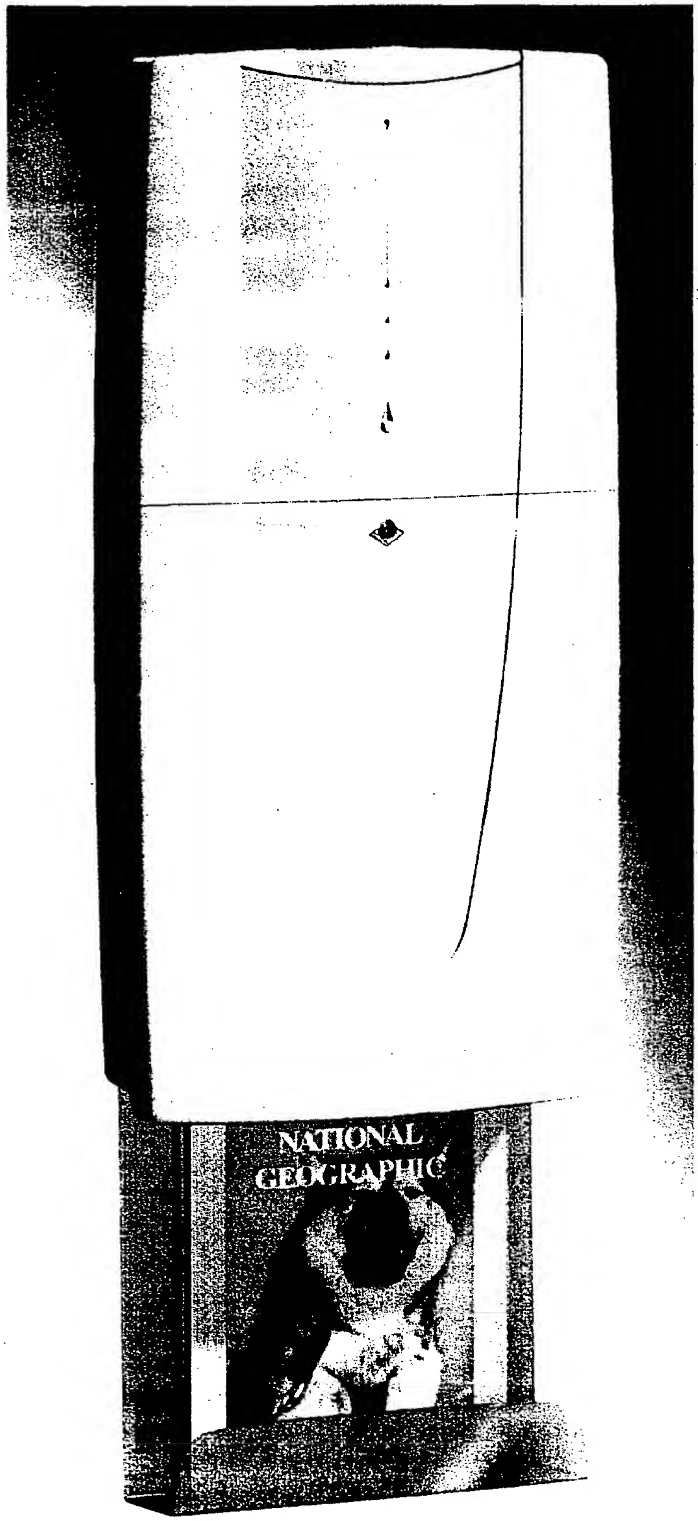


Figure 1. Wallprinter

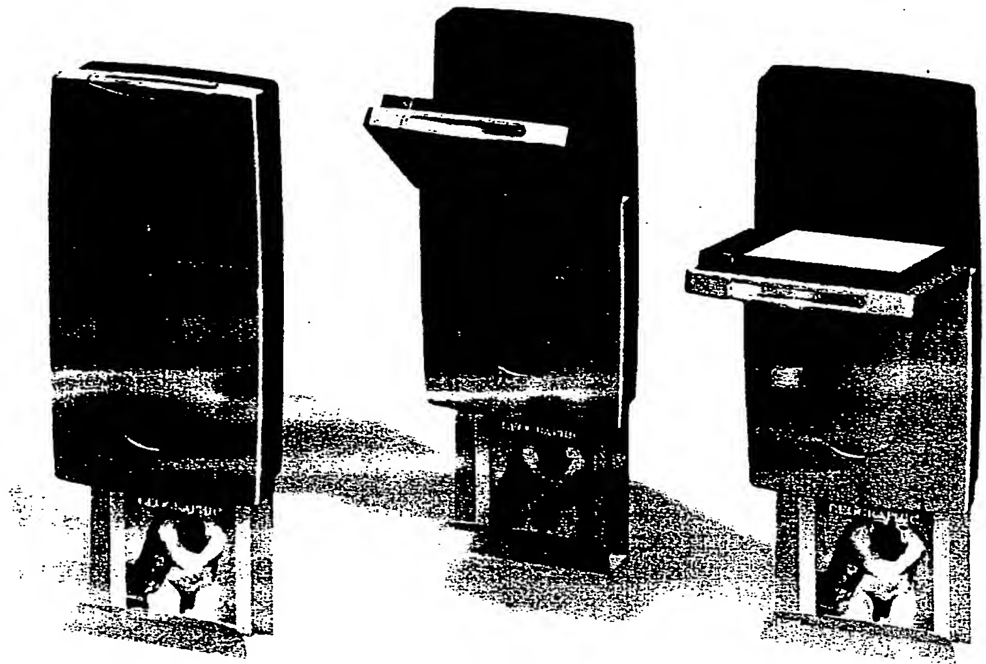


Figure 2. Wallprinter paper and ink cartridge access

The printer has several metal hangers on the rear, which locate into keyhole slots in a metal back plate that is securely fastened to a wall. When the printer has been connected, it is hung onto the back plate and fixed with a locking screw found under the paper tray.

The Netpage Printers are fully customizable in finishes and color as the front moldings clip on to a core chassis and are easily removable.

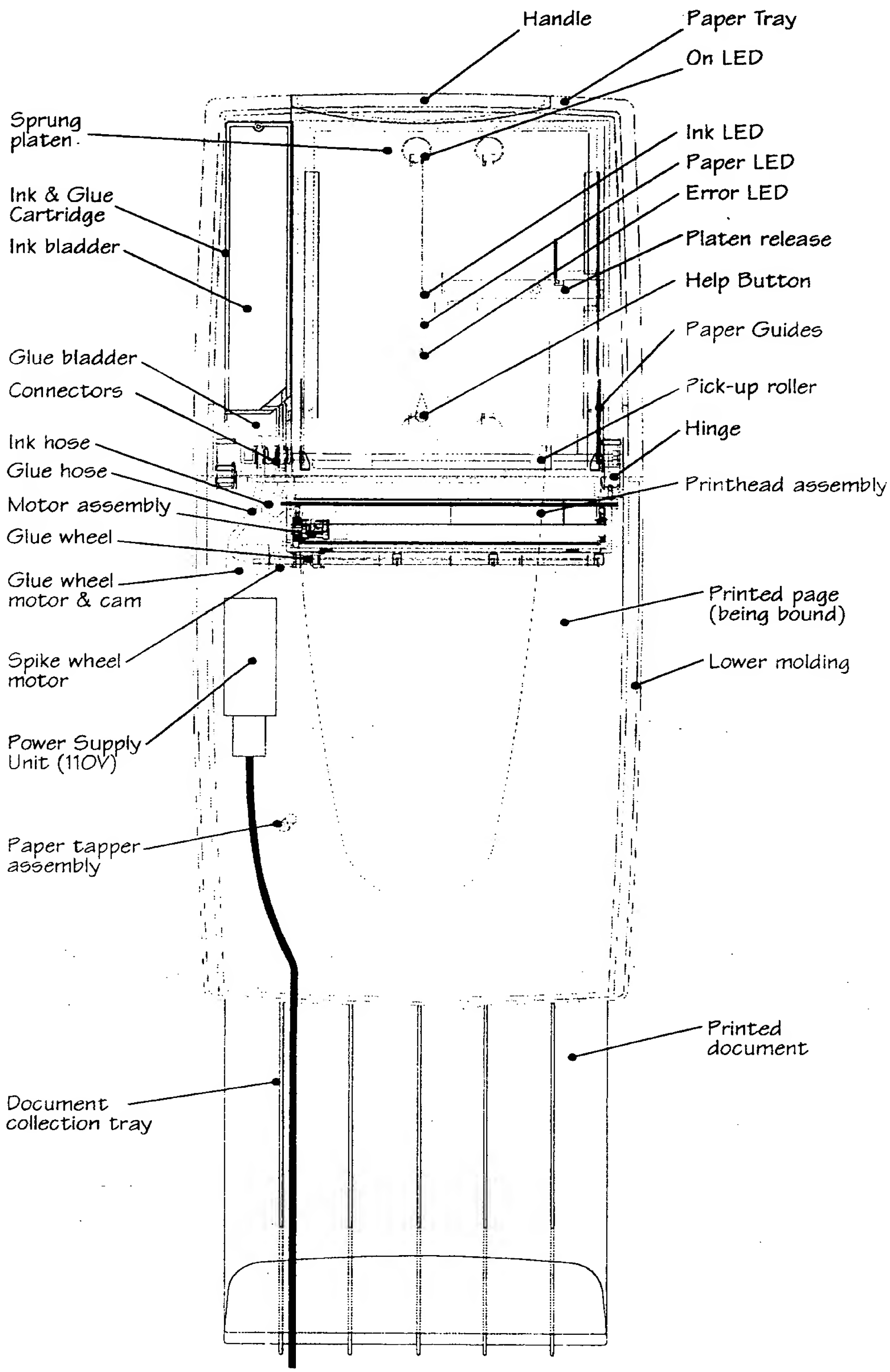


Figure 3. Wallprinter front elevation

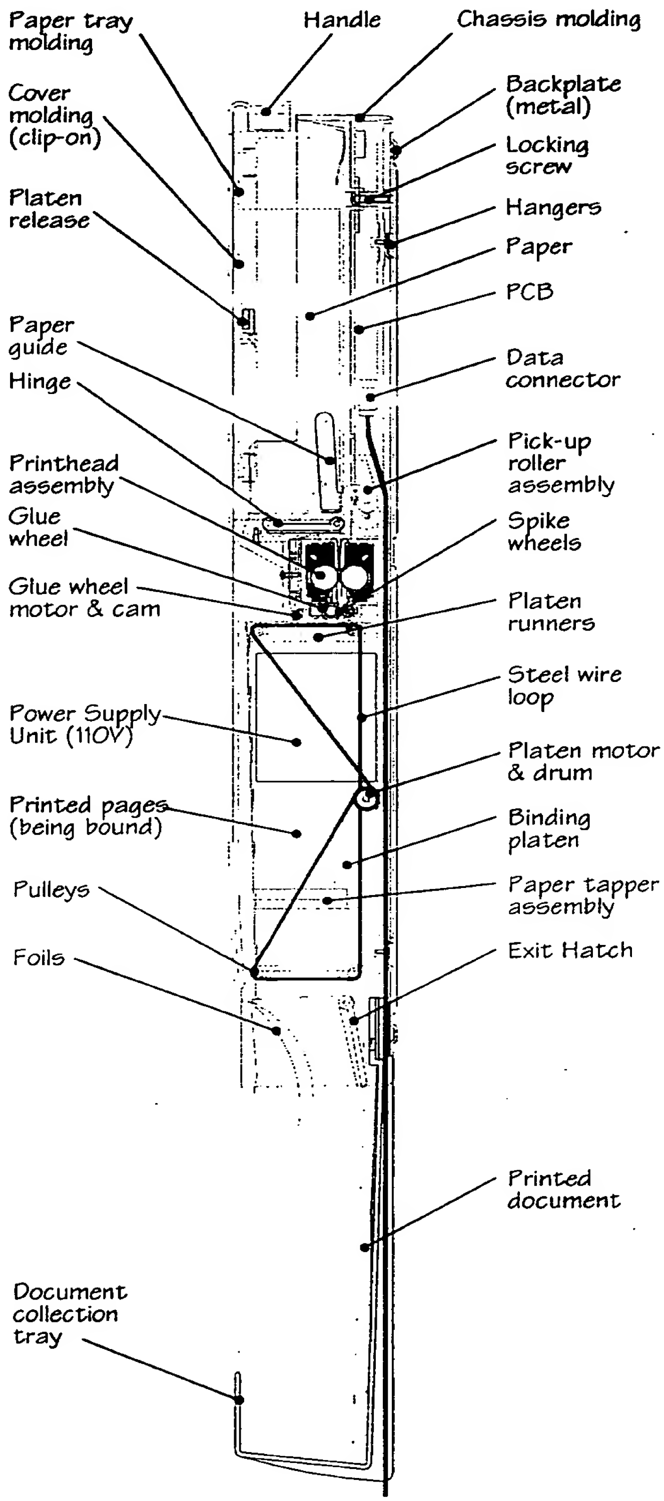


Figure 4. Wallprinter side elevation

2.2 WALLPRINTER PRO

This printer is similar to Wallprinter in most respects, except that it has a duplex 11" printhead assembly, which prints on US Letter paper in a landscape format (see Figures 5, 6 and 7). This means a faster print time and binding time for each page, making for faster overall document delivery.

Another difference is the location of the ink cartridge, which resides above the paper tray rather than down the side. Each page is glued along the horizontal edge by a full-length glue sponge, which is capped when not in use. Operation, printing, and document handling are identical to Wallprinter.

Wallprinter Pro is fully customizable in finishes and color.

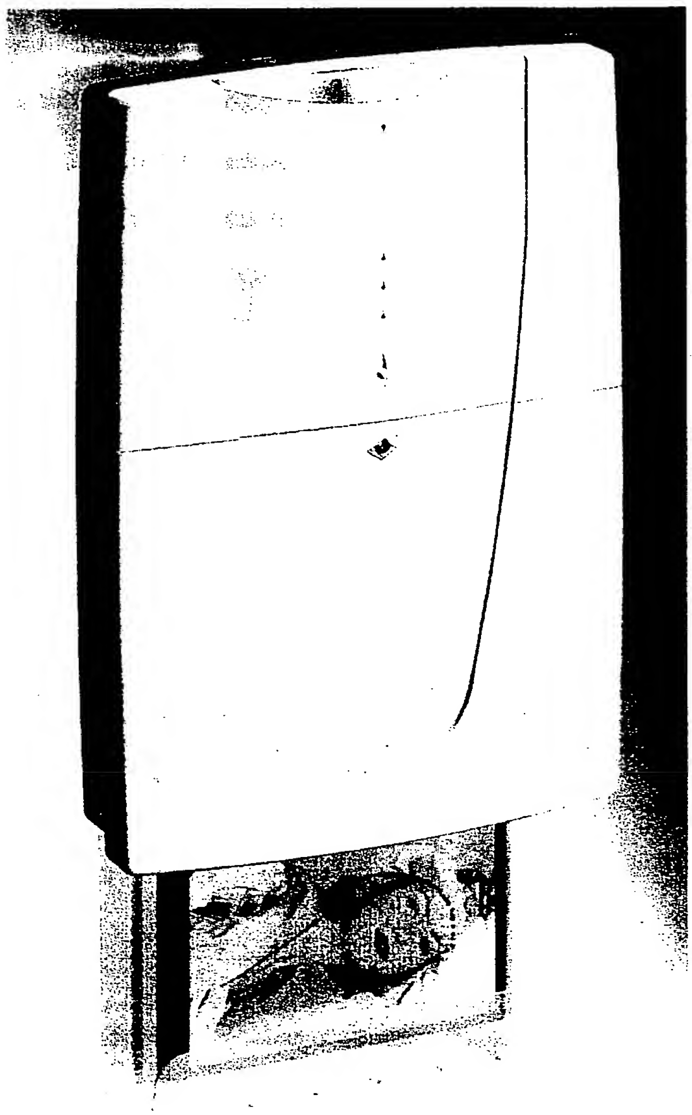


Figure 5. Wallprinter Pro

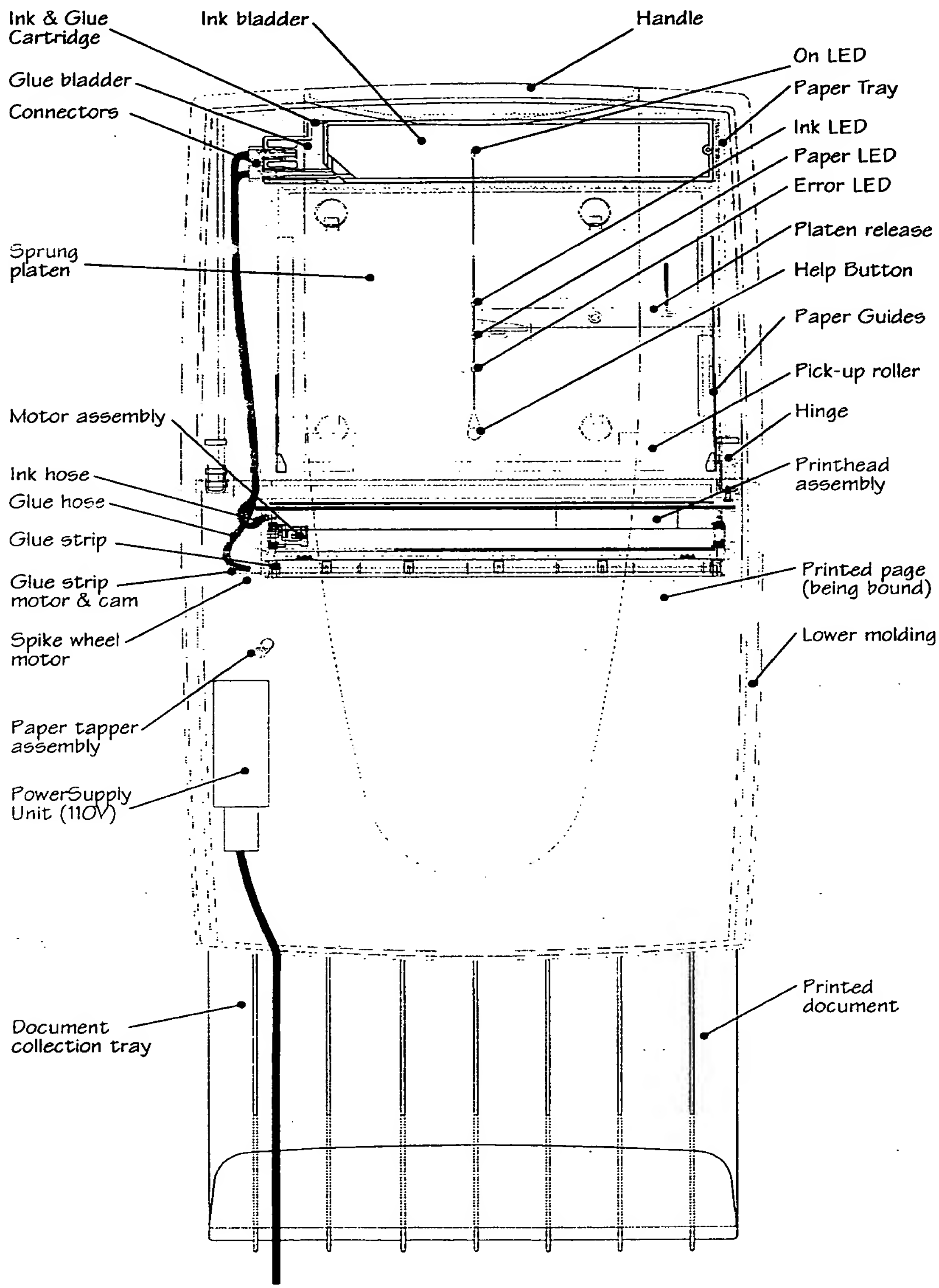


Figure 6. Wallprinter Pro front elevation

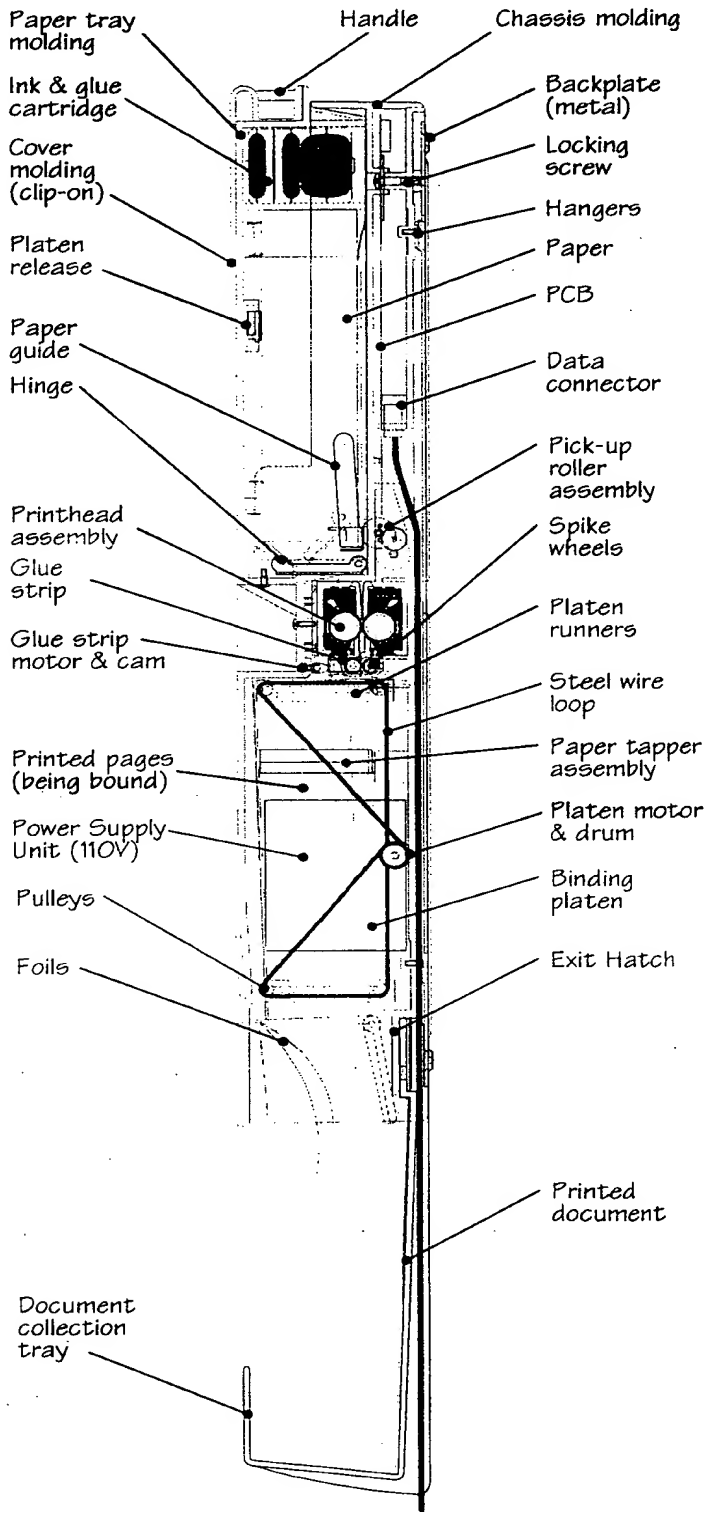


Figure 7. Wallprinter Pro side elevation

2.3 WALLPRINTER PRO R

This printer shares the same printing and binding configuration of US Letter landscape format as Wallprinter Pro. The main difference is the media delivery, which is in the form of a large print cartridge cartridge (see Figures 8, 9 and 10). This cartridge accommodates C, M, Y, and infrared inks and glue as well as a 1000 sheet capacity roll of paper. The cartridge can be recharged at nominated outlets when required and it is protected from forgeries by an authentication chip [1,2].

The printer has integral structural metalwork to support its weight and a ball bearing track for easy loading and removal.

Wallprinter Pro R is fully customizable in finishes and color.

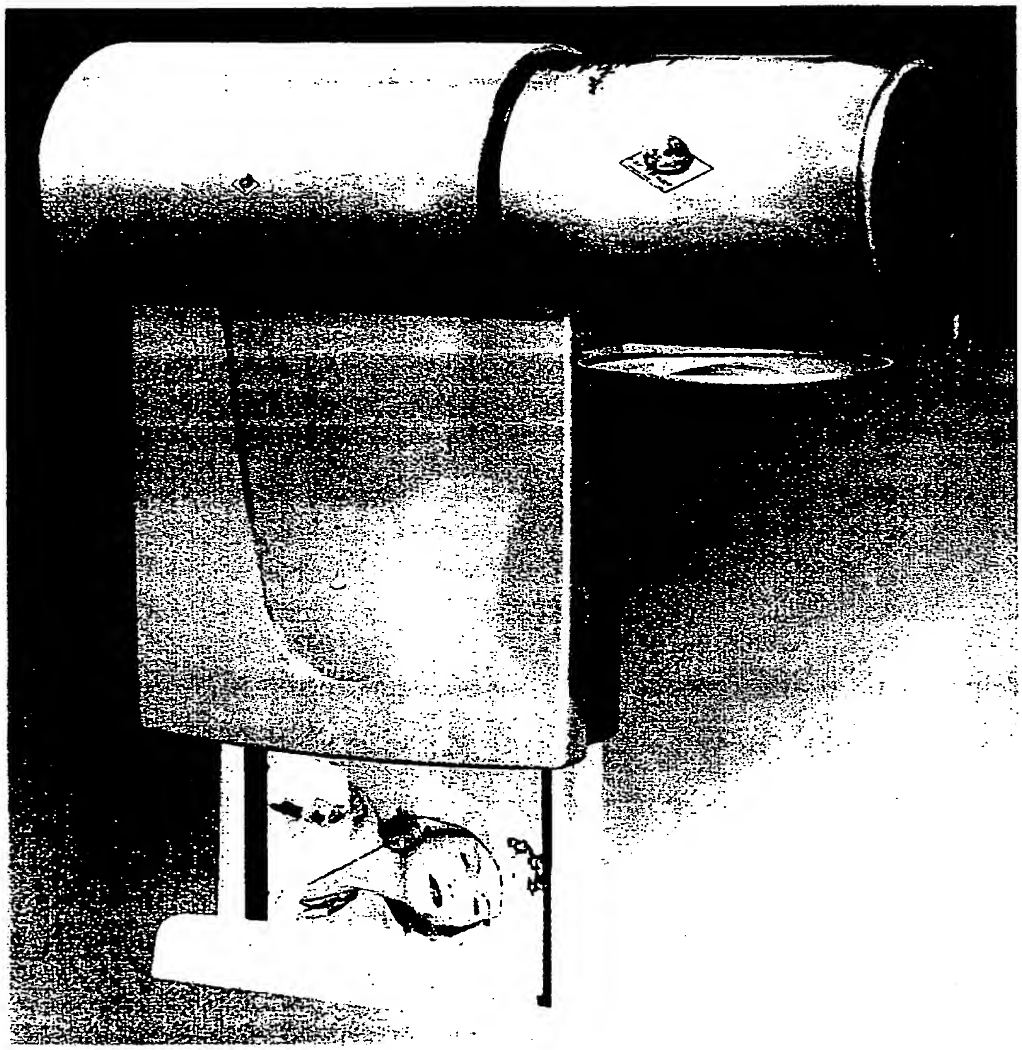


Figure 8. Wallprinter Pro R, with print cartridge extracted

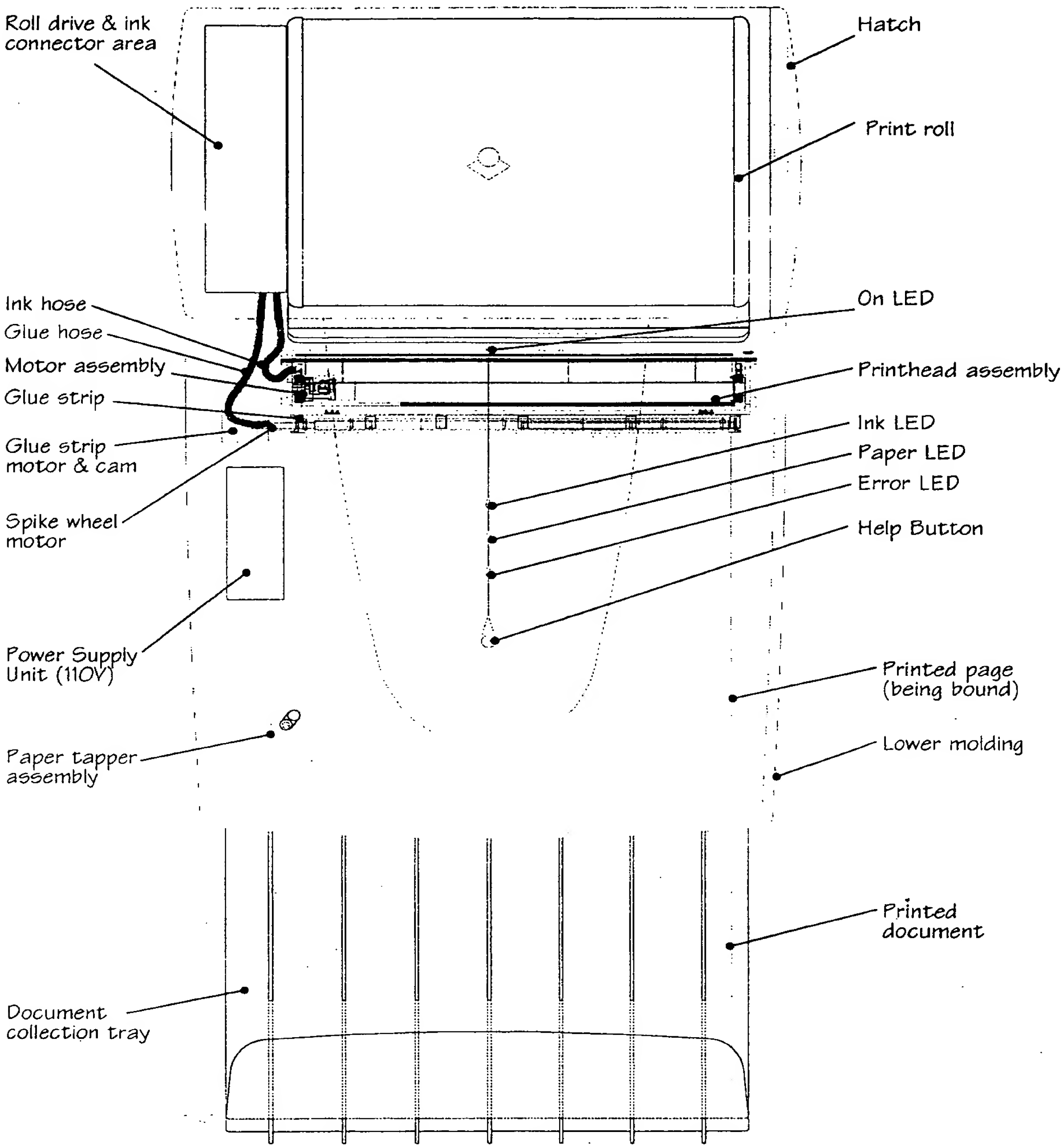


Figure 9. Wallprinter Pro R front elevation

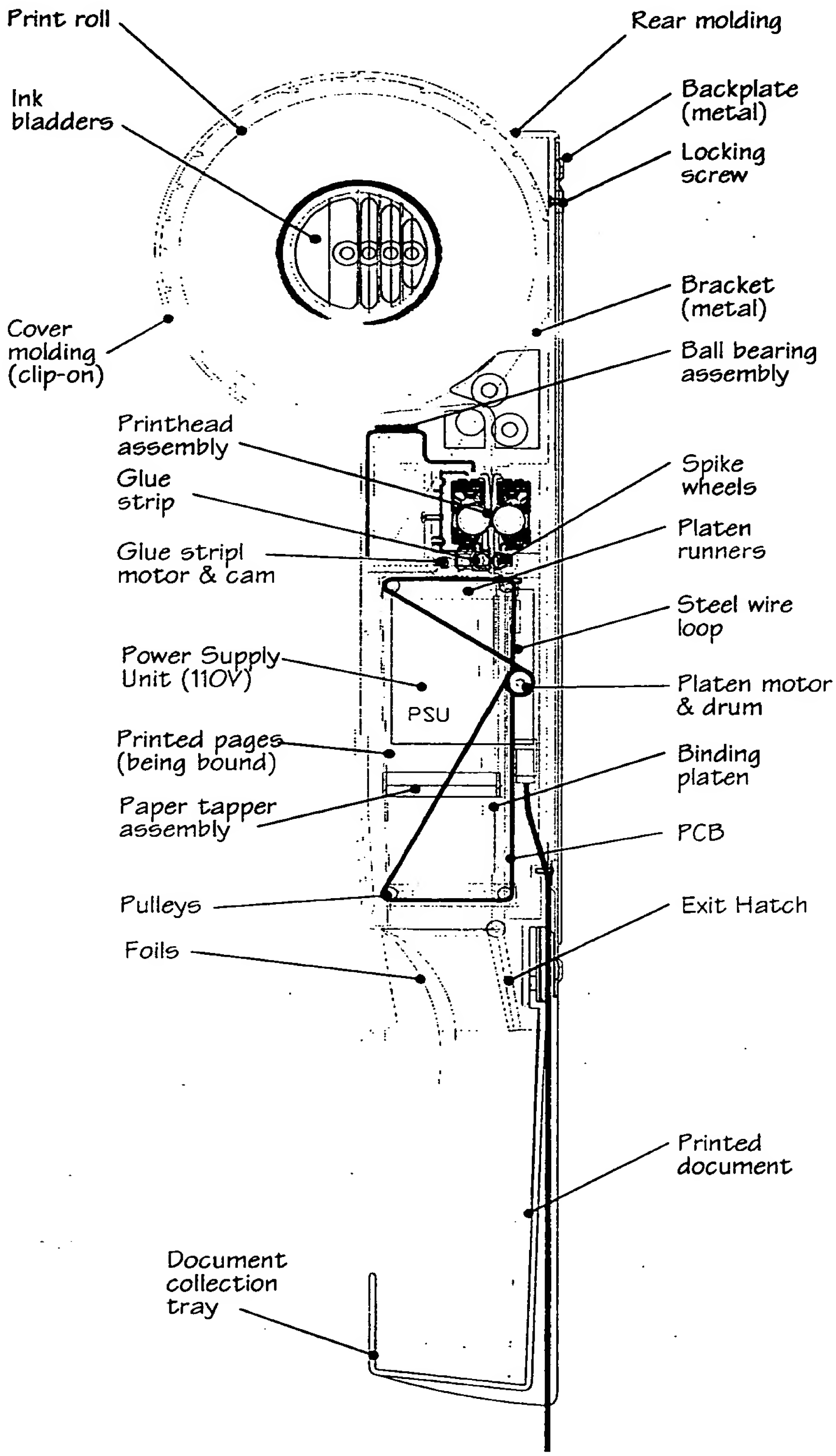


Figure 10. Wallprinter Pro R side elevation

3 Tableprinter

3.1 TABLEPRINTER PRO

This printer is a tabletop version of the Wallprinter Pro. Essentially, it is the same printer unit with a base plinth that adds extra functionality, such as USB, parallel port and a power socket (see Figure 11).

Tableprinter Pro is fully customizable in finishes and color.

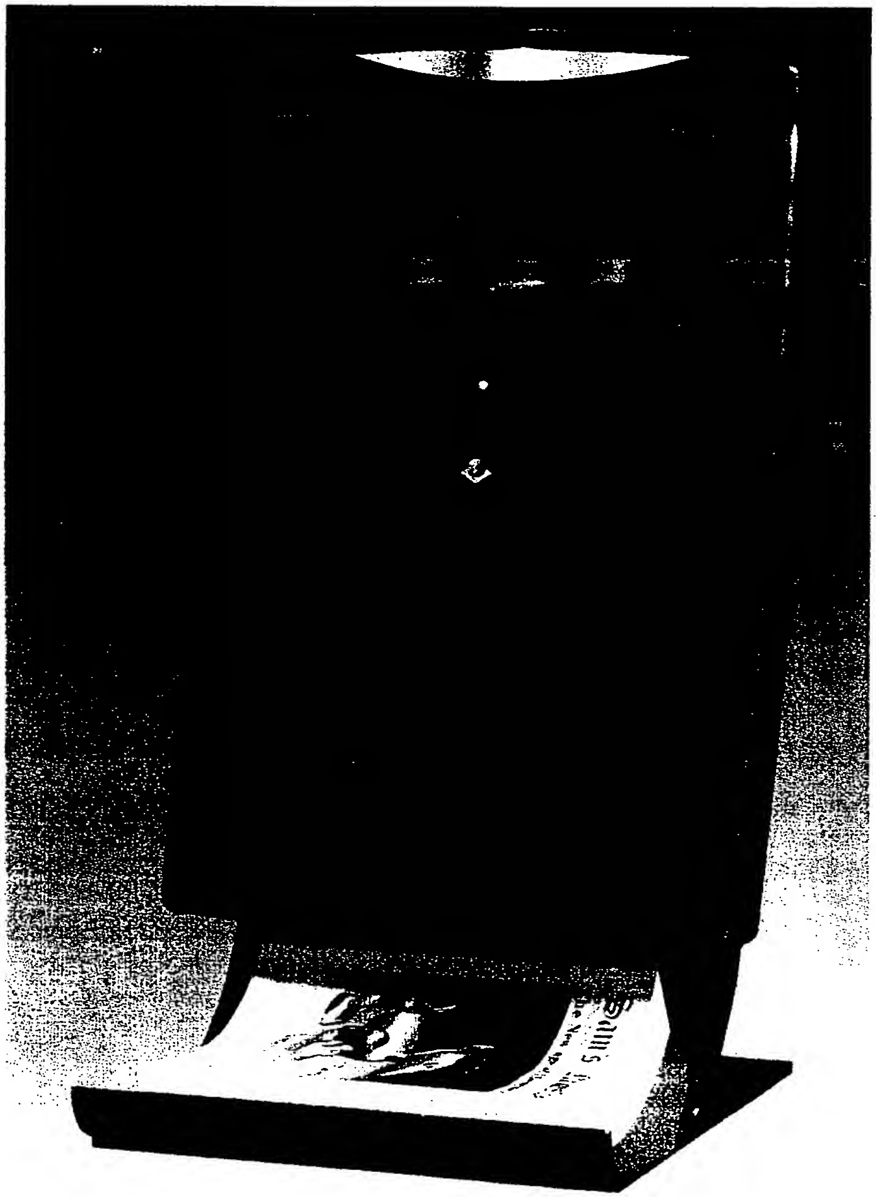


Figure 11. Tableprinter Pro

3.2 TABLEPRINTER PRO R

This printer is a tabletop version of the Wallprinter Pro R. Essentially, it is the same printer unit with a base plinth that adds extra functionality, such as USB, parallel port and a power socket (see Figure 12).

Tableprinter Pro R is fully customizable in finishes and color.

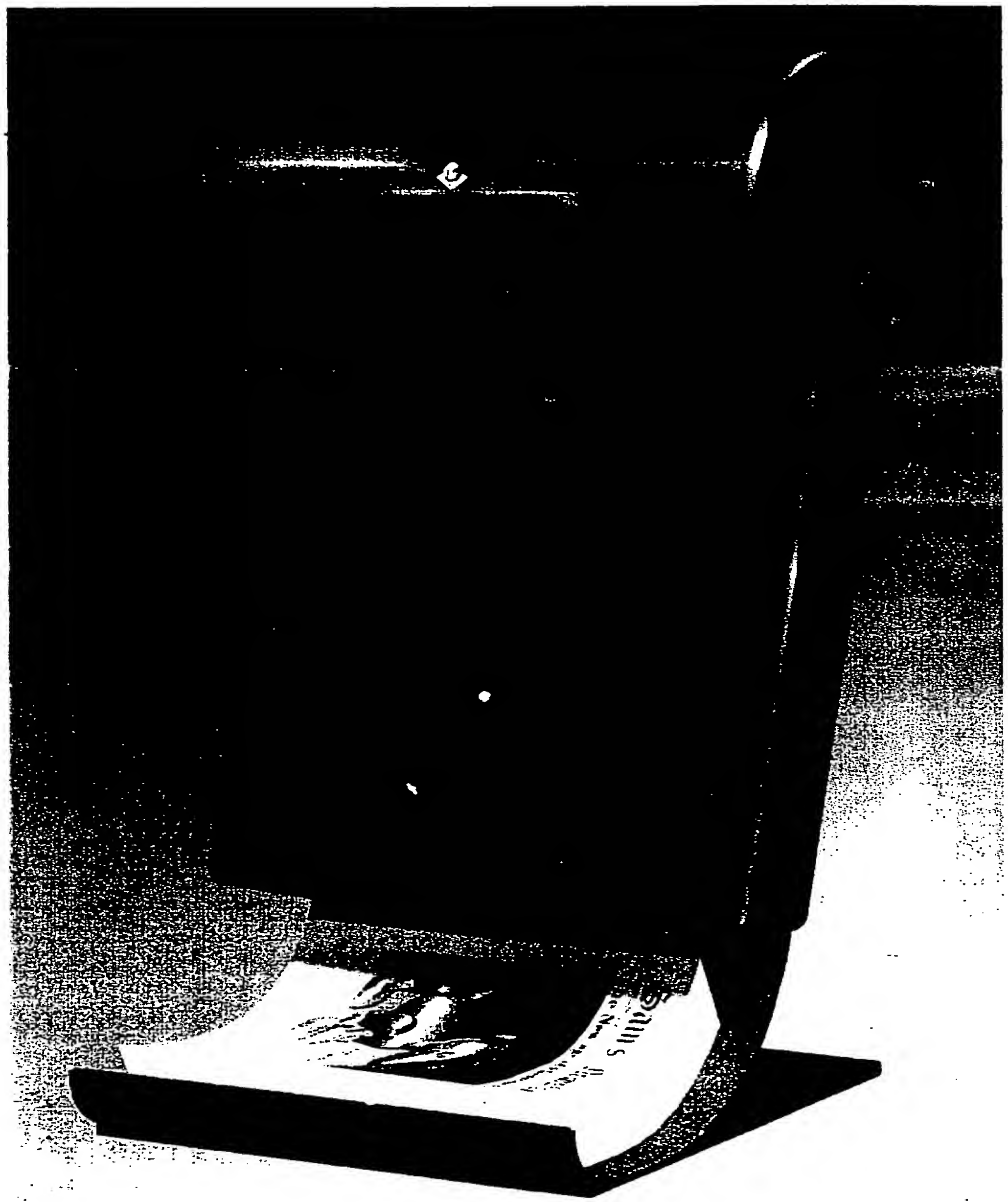


Figure 12. Tableprinter Pro R

4 Deskprinter

4.1 Deskprinter Pro R

The Deskprinter Pro R is a vertical format printer based around the large format print cartridge (see Figures 13, 14, 15 and 16). This cartridge accommodates C, M, Y, Infrared inks and glue, as well as a 1000 page roll of paper. The print cartridge is lowered into the top of the unit through a latching hatch. When printing, the paper is fed through the printhead assembly where it is duplex printed and then ejected past a cutter which crops it into sheets. As the sheet passes the cutter, a glue wheel assembly glues along the long edge and powered spike rollers propel the sheet out into the binding area.

The binding area consists of a binding platen which operates with a closed steel wire loop system of pulleys, runners and a powered axle. This provides the necessary speed to push the glued sheet forward onto the rear of a previous sheet, glue/bind it and return to the home position to accept the next printed sheet in less than 2 seconds. A motorized paper tapper assembly aligns the sheets in a simultaneous operation.

When the document has been bound, a series of metal fingers at the top and bottom of the document rotate out of the way and the document is pushed into the out tray area. A powered hatch opens and a motorized ejector assembly pushes the document through the exit slot of the unit. The hatch then closes onto the document ready for removal. Subsequent documents are stacked up in a similar fashion. The user interface is identical to the other printers in the range. A large PCB contains all the processing and external interface components and connects to an internal 110V/220V power supply unit.

Deskprinter Pro R is fully customizable in finishes and color.

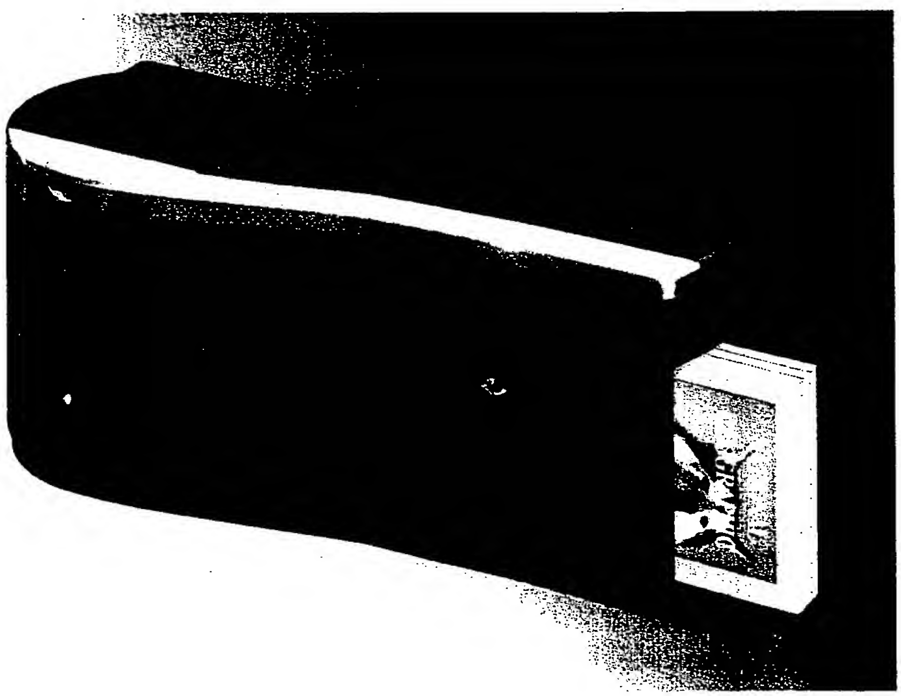


Figure 13. Deskprinter Pro R

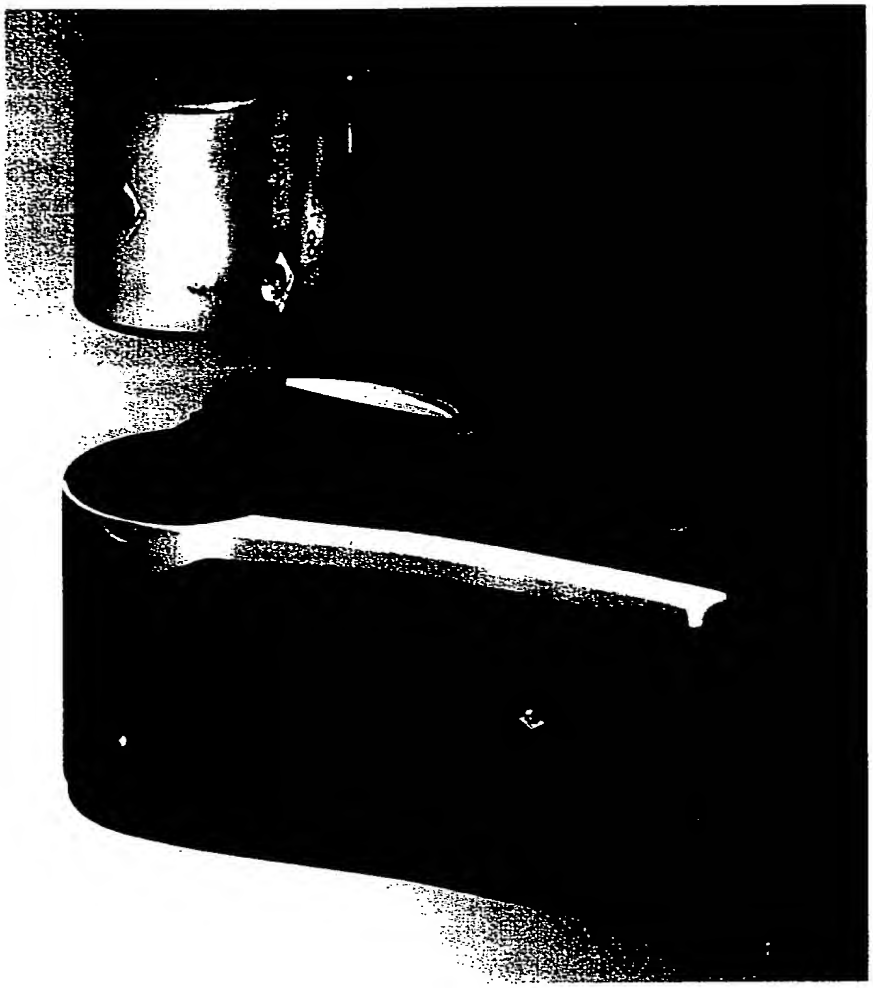


Figure 14. Deskprinter Pro R, with print cartridge extracted

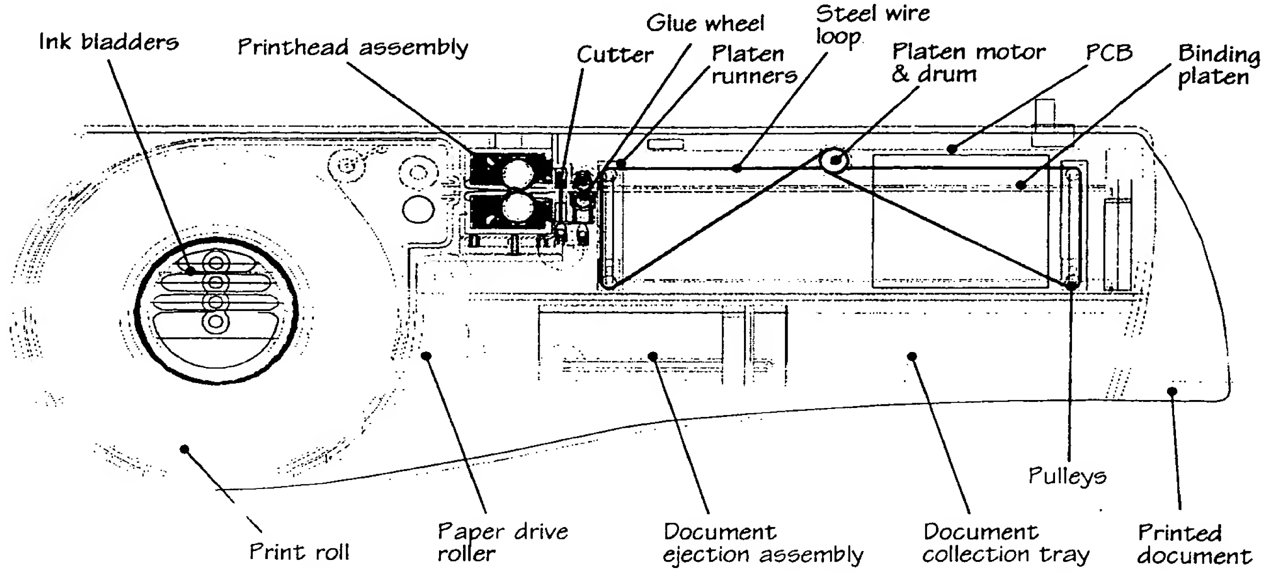


Figure 15. Deskprinter Pro R plan

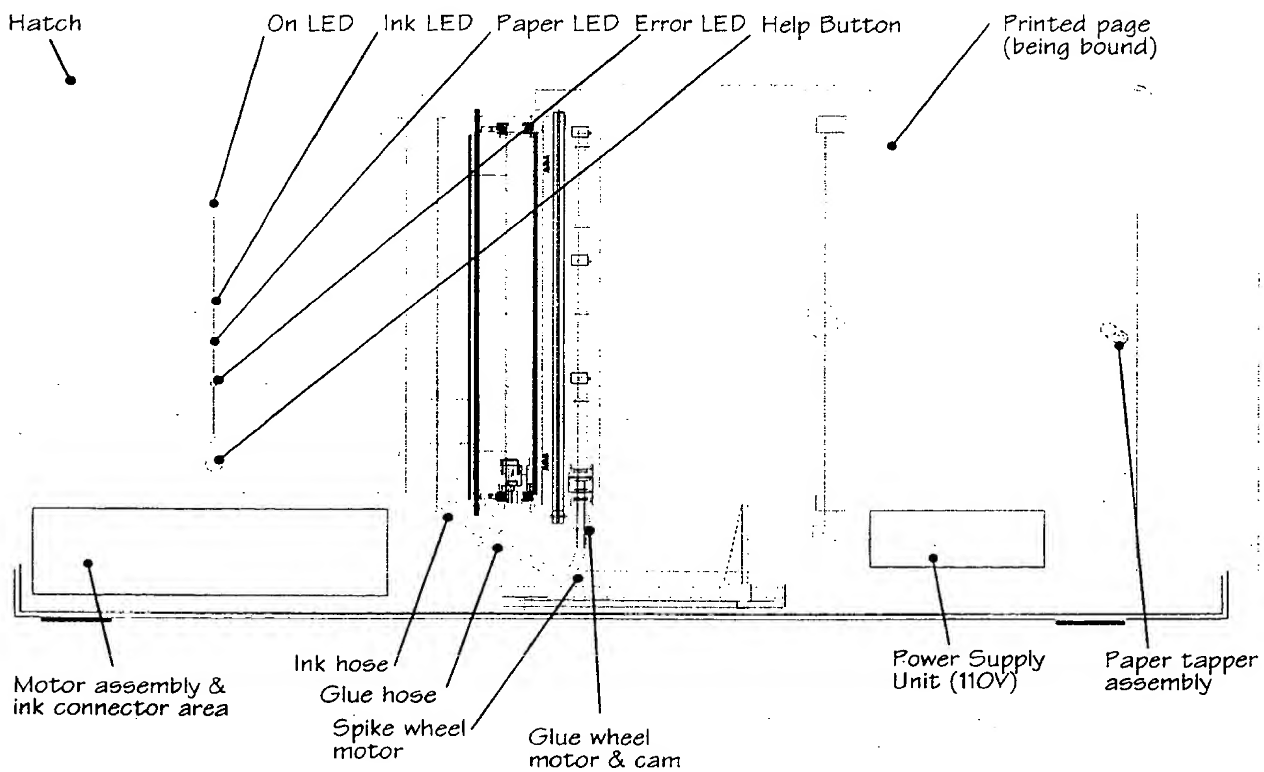


Figure 16. Deskprinter Pro R elevation

5 Travelprinter

5.1 TRAVELPRINTER R

The Travelprinter R is a small, lightweight, versatile and completely portable Netpage Printer (see Figures 17, 19, 20 and 21). It has in-built mobile network communication hardware and software, allowing it to download documents anywhere. Travelprinter also has communications ports for computer interface printing when required.

The printer consists of a front and rear molding with a chassis to accommodate the major components including a lithium battery and an 8½" duplex Memjet printhead assembly.

A compact print cartridge cartridge with C, M, Y and infrared inks and paper is used in the printer, providing 50 US Letter pages or 100 A5 pages. It is protected from forgeries by an authentication chip.

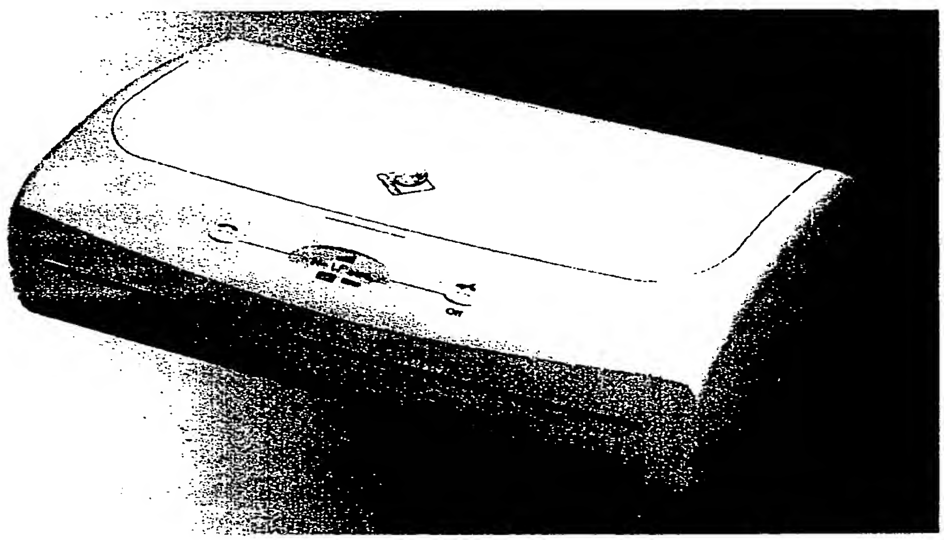


Figure 17. Travelprinter R

A motorized guillotine assembly cuts the media between the cartridge and the printhead and motorized spike wheels eject the finished print out of the unit. A flex PCB runs from the main board to a segment LCD and two push buttons. The LCD shows signal strength, any errors, battery and number of pages left in the cartridge. The buttons allow the printer to either connect to the Netpage Network or to act as a stand-alone printer. A USB interface is provided on the side of the printer along with DC 3V input.

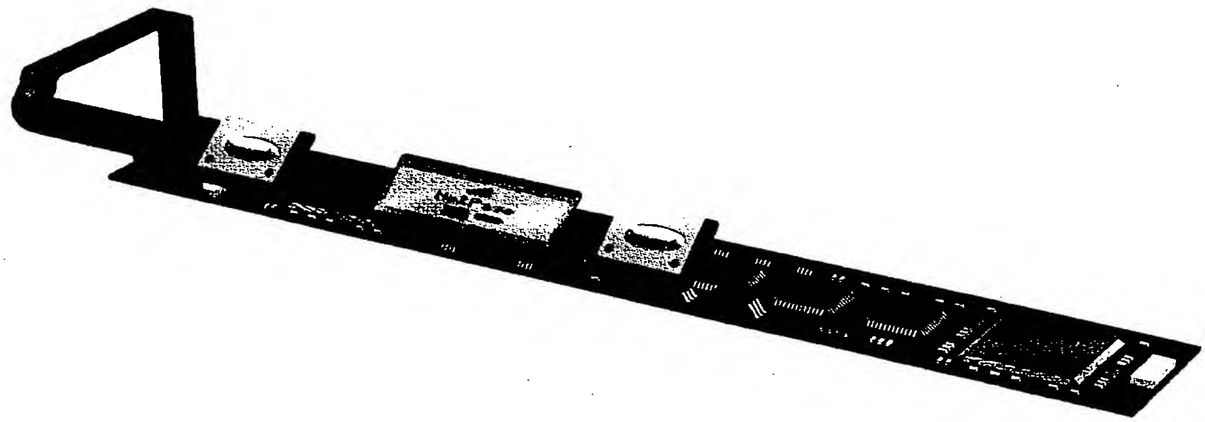


Figure 18. Travelprinter R main and flex PCB, with buttons and segment LCD

Figure 19. Travelprinter R, with print cartridge extracted

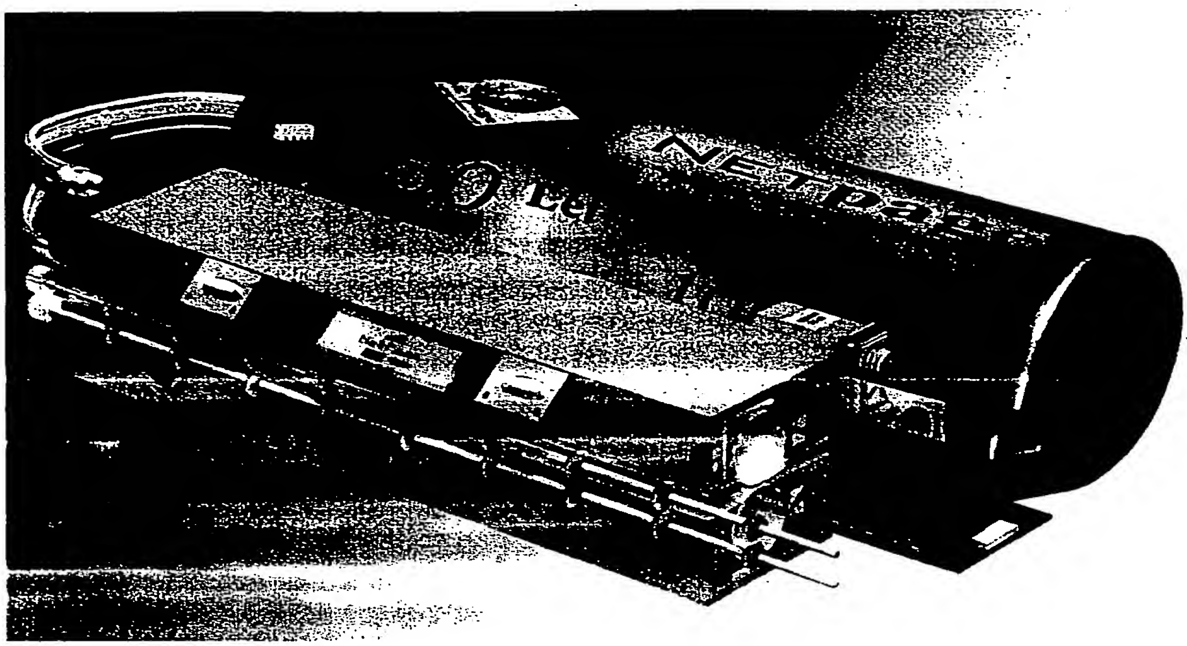


Figure 20. Travelprinter R detail, showing duplexed imaging units and print cartridge

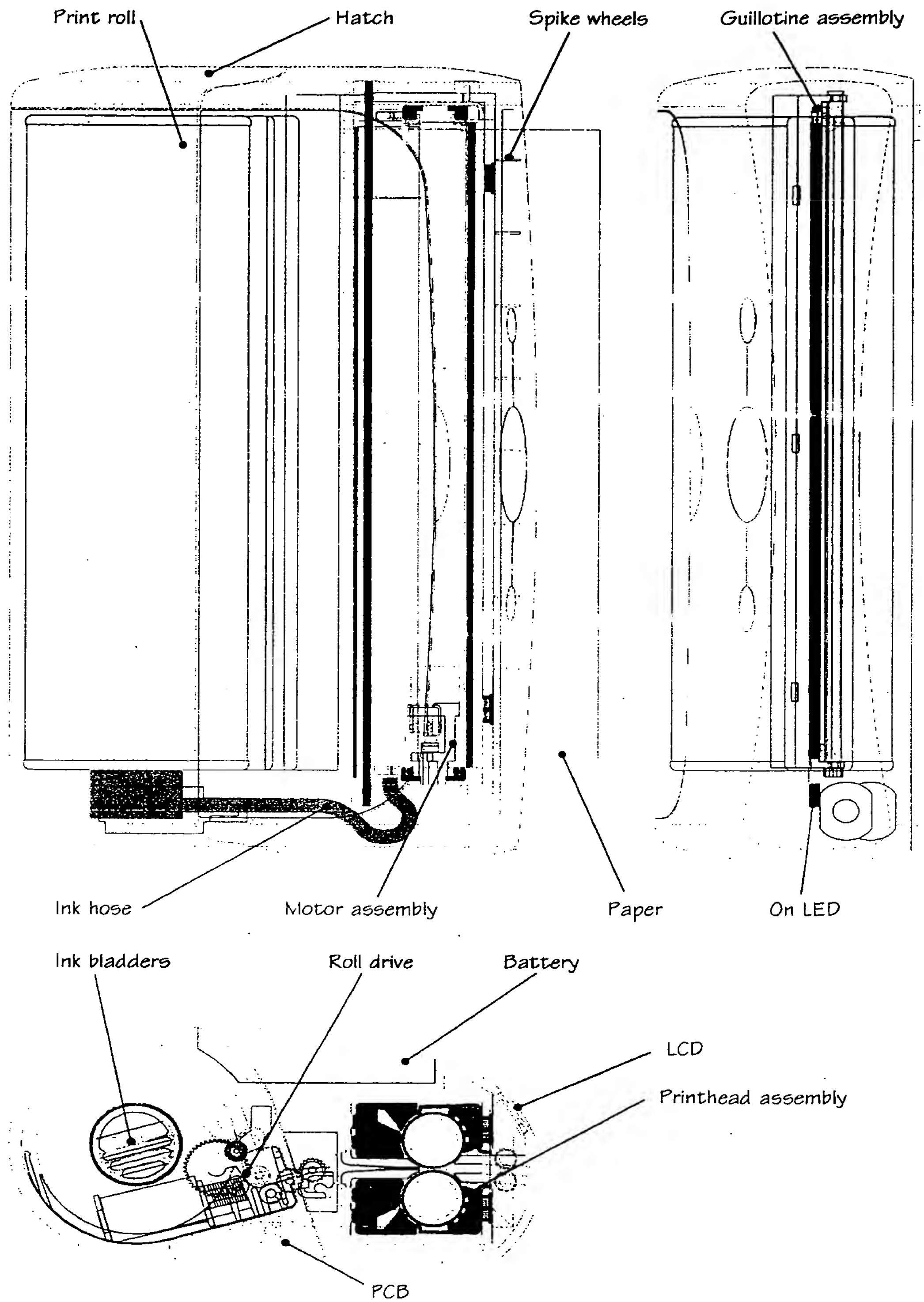


Figure 21. Travelprinter plan and elevations

6 Microprinter

6.1 MICROPRINTER

The Microprinter is a small, versatile, pocket-size printer/camera (see Figures 22 and 23). It has built-in mobile network communication hardware and software, allowing it to link to the Netpage Network and fetch documents from anywhere. In addition, the product is ergonomically configured and styled as a fully functional digital camera with the standard Classic, HDTV and Panoramic print formats for photography.

The Microprinter accommodates a 4" (100mm) page-width print cartridge with C, M, Y and infrared ink plus 5.4 meters of paper. This means the user can single-sided print 41 Netpages, 36 Classic, 30 Horizontal or 18 Panoramic prints or any combination thereof.

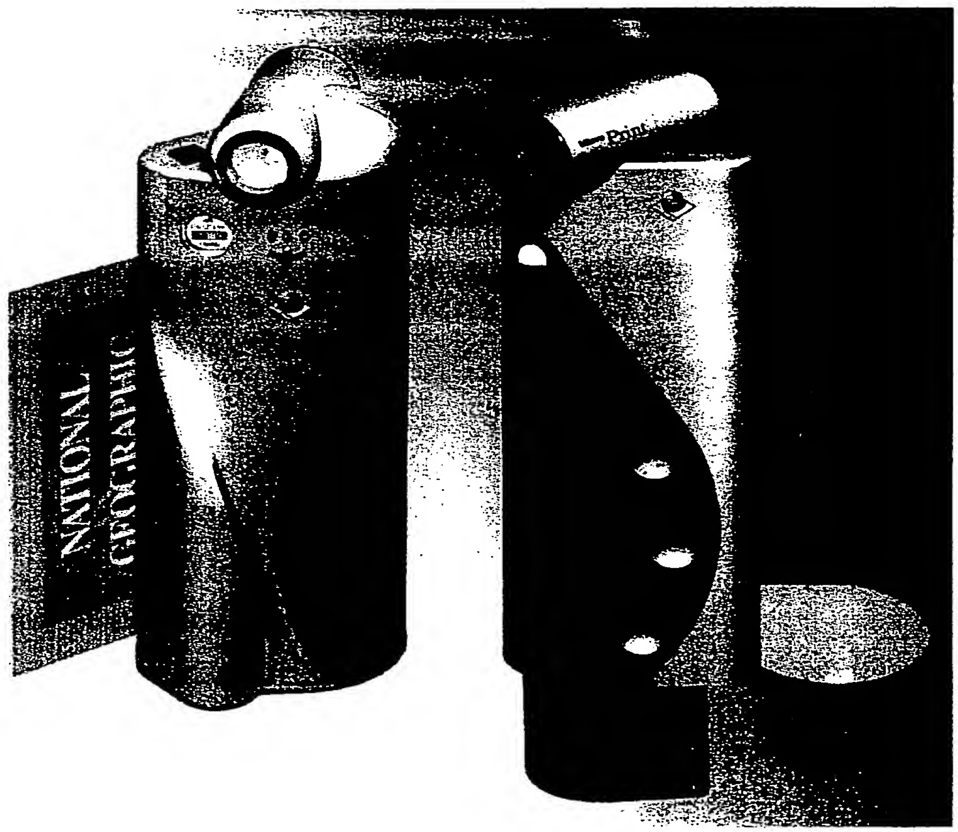


Figure 22. Microprinter rear view with printing in progress (left), and front view with print cartridge extracted (right)

The Microprinter is a fixed-lens camera with auto focus and digital photo enhancement capabilities, which allows the user to take sophisticated photos very simply and easily. A print button allows the user to print and reprint a photo or Netpage.

The Microprinter consists of two main front and rear moldings and a hatch. The main moldings accommodate the lens assembly, viewfinder optics and a rigid PCB. A flex PCB runs from the main board to the imaging chip, an LCD, the printhead assembly, the print-

head capping mechanism, a cutter assembly, the roll motor drive, various control switches and a CR2 battery.

The Microprinter is fully customizable in finishes and color.

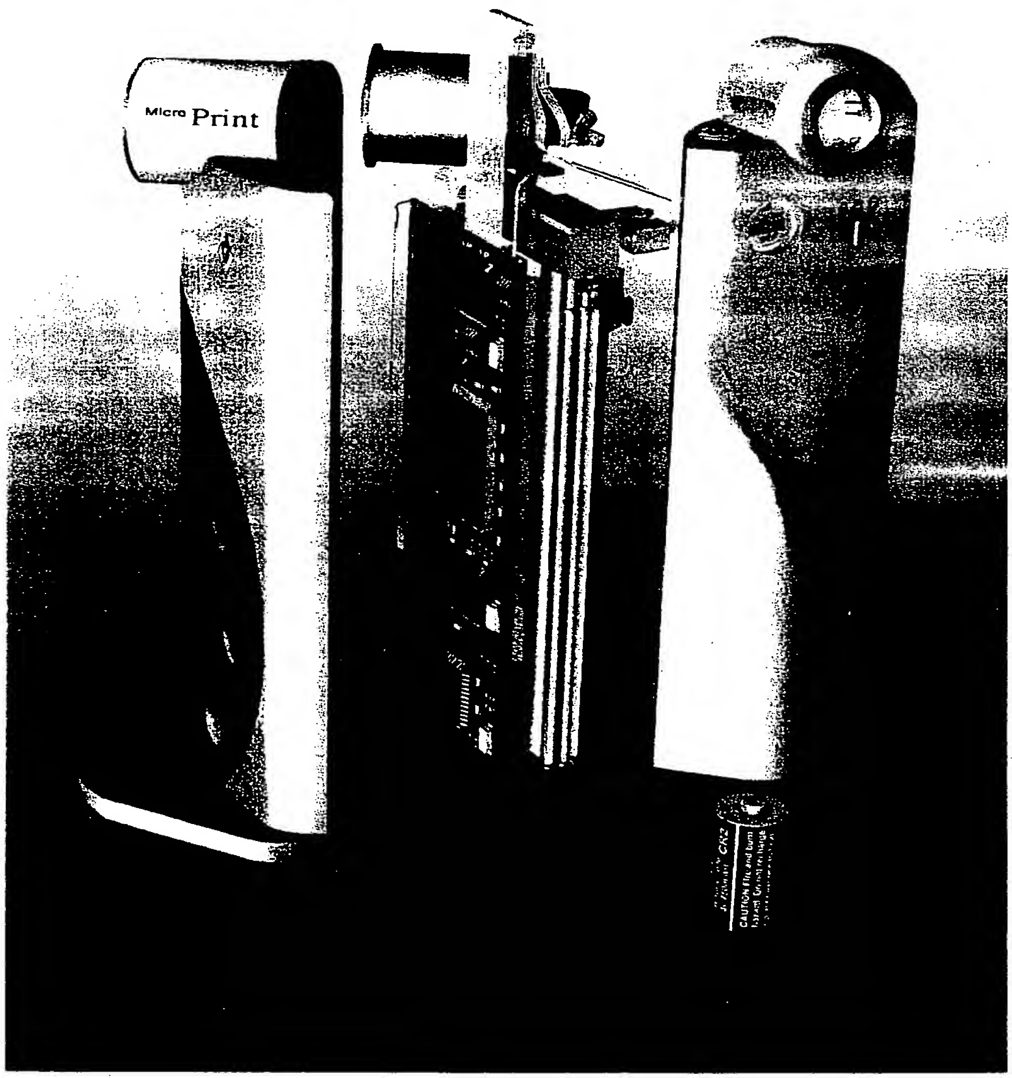


Figure 23. Microprinter exploded view

PRINTER CONTROLLER

7 Memjet-Based Printing

A Memjet printhead produces 1600 dpi bi-level CMYK. On low-diffusion paper, each ejected drop forms an almost perfectly circular 22.5μm diameter dot. Dots are easily produced in isolation, allowing dispersed-dot dithering to be exploited to its fullest. Since the Memjet printhead is the width of the page and operates with a constant paper velocity, the four color planes are printed in perfect registration, allowing ideal dot-on-dot printing. Since there is consequently no spatial interaction between color planes, the same dither matrix is used for each color plane. Dot-on-dot printing minimizes 'muddying' of midtones caused by inter-color bleed.

A page layout may contain a mixture of images, graphics and text. Continuous-tone (contone) images and graphics are reproduced using a stochastic dispersed-dot dither. Unlike a clustered-dot (or amplitude-modulated) dither, a *dispersed-dot* (or frequency-modulated) dither reproduces high spatial frequencies (i.e. image detail) almost to the limits of the dot resolution, while simultaneously reproducing lower spatial frequencies to their full color depth, when spatially integrated by the eye. A *stochastic* dither matrix is carefully designed to be free of objectionable low-frequency patterns when tiled across the image. As such its size typically exceeds the minimum size required to support a particular number of intensity levels (e.g. 16×16×8 bits for 257 intensity levels).

Human contrast sensitivity peaks at a spatial frequency of about 3 cycles per degree of visual field and then falls off logarithmically, decreasing by a factor of 100 beyond about 40 cycles per degree and becoming immeasurable beyond 60 cycles per degree [7,8]. At a normal viewing distance of 12 inches (about 300mm), this translates roughly to 200-300 cycles per inch (cpi) on the printed page, or 400-600 samples per inch according to Nyquist's theorem.

In practice, contone resolution above about 300 ppi is of limited utility outside special applications such as medical imaging. Offset printing of magazines, for example, uses contone resolutions in the range 150 to 300 ppi. Higher resolutions contribute slightly to color error through the dither.

Black text and graphics are reproduced directly using bi-level black dots, and are therefore not antialiased (i.e. low-pass filtered) before being printed. Text is therefore *supersampled* beyond the perceptual limits discussed above, to produce smoother edges when spatially integrated by the eye. Text resolution up to about 1200 dpi continues to contribute to perceived text sharpness (assuming low-diffusion paper, of course).

The Netpage Printer uses a contone resolution of 267 ppi (i.e. 1600 dpi / 6), and a black text and graphics resolution of 800 dpi.

8 Document Data Flow

Because of the page-width nature of the Memjet printhead, each page must be printed at a constant speed to avoid creating visible artifacts. This means that the printing speed can't be varied to match the input data rate. Document rasterization and document printing are therefore decoupled to ensure the printhead has a constant supply of data. A page is never printed until it is fully rasterized. This is achieved by storing a compressed version of each rasterized page image in memory.

This decoupling also allows the RIP to run ahead of the printer when rasterizing simple pages, buying time to rasterize more complex pages.

Because contone color images are reproduced by stochastic dithering, but black text and line graphics are reproduced directly using dots, the compressed page image format contains a separate foreground bi-level black layer and background contone color layer. The black layer is composited over the contone layer after the contone layer is dithered.

Figure 24 shows the flow of a Netpage Printer document from network to printed page.

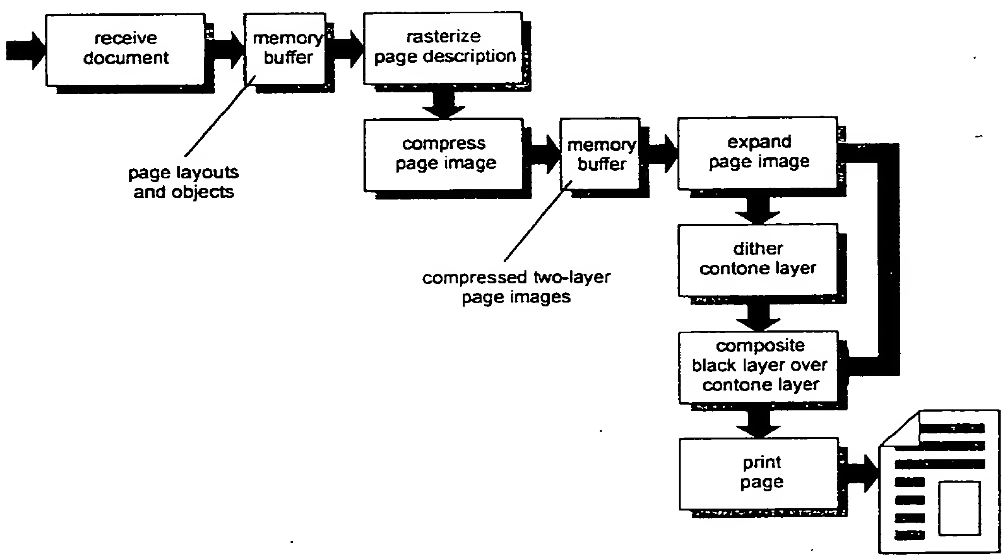


Figure 24. Netpage Printer document data flow

At 267 ppi, a Letter page of contone RGB or CMY data has a size of 19MB. Using lossy contone compression algorithms such as JPEG [9], contone images compress with a ratio up to 10:1 without noticeable loss of quality, giving a compressed page size of 1.9MB.

At 800 dpi, a Letter page of bi-level data has a size of 7MB. Coherent data such as text compresses very well. Using lossless bi-level compression algorithms such as Group 4 Facsimile [6], ten-point text compresses with a ratio of about 10:1 (as discussed in Section 11.2.1.2), giving a compressed page size of 0.8MB.

Once dithered, a page of CMY contone image data consists of 86MB of bi-level data. Using lossless bi-level compression algorithms on this data is pointless precisely because the optimal dither is stochastic - i.e. since it introduces hard-to-compress disorder.

The two-layer compressed page image format therefore exploits the relative strengths of lossy JPEG contone image compression and lossless bi-level text compression. The format is compact enough to be storage-efficient, and simple enough to allow straightforward real-time expansion during printing.

Since text and images normally don't overlap, the normal worst-case page image size is 1.9MB (i.e. image only), while the normal best-case page image size is 0.8MB (i.e. text only). The absolute worst-case page image size is 2.7MB (i.e. text over image). Assuming a quarter of an average page contains images, the average page image size is 1.1MB.

9 Printer Controller Architecture

The Netpage Printer controller consists of a controlling processor, a factory-selected network interface, a radio transceiver, dual raster image processor (RIP) DSPs, duplexed print engines, flash memory, and 64MB of DRAM, as illustrated in Figure 25.

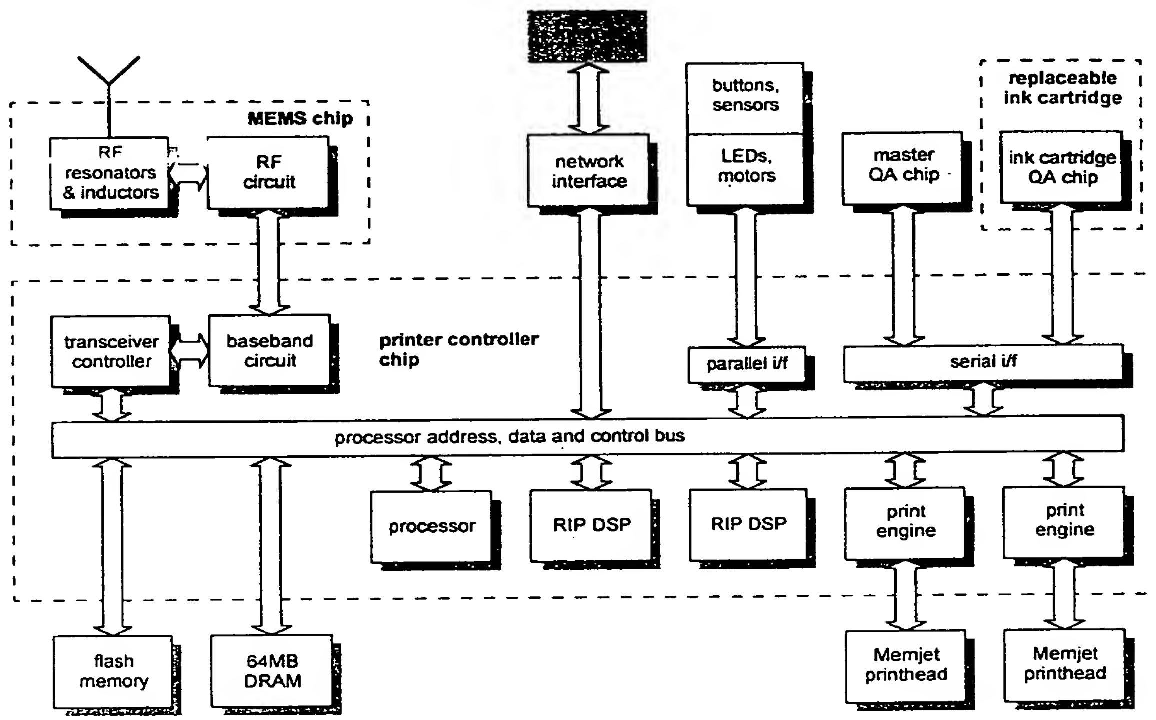


Figure 25. Basic printer controller architecture

The controlling processor handles communication with the Internet and with local wireless pens, controls the user interface (buttons and LEDs), controls the paper transport, handles ink cartridge authentication and ink monitoring, and feeds and synchronizes the RIP DSPs and print engines. It consists of a medium-performance general-purpose microprocessor.

The RIP DSPs rasterize and compress page descriptions to the Netpage Printer's compressed page format. Each print engine expands, dithers and prints page images to its associated Memjet printhead in real time (i.e. at 30 or 45 pages per minute). The duplexed print engines print both sides of the page simultaneously.

The printer controller's flash memory holds the software for both the processor and the DSPs, as well as configuration data. This is copied to main memory at boot time.

The processor, DSPs, print engines and digital transceiver components are integrated in a single ASIC. The MEMS and analog RF components are integrated in a separate MEMS chip, which is also used in the wireless pen. Additional pen-specific components in the

MEMS chip are not used in the printer controller. The Internet network interface module is separate, since Netpage Printers allow the network connection to be factory-selected. Flash memory and the 2x256Mbit (64MB) DRAM is also off-chip.

Various Internet network interface modules can be supported. Possibilities include a POTS modem, a Hybrid Fiber-Coax (HFC) cable modem, an ISDN modem, a DSL modem, a satellite transceiver, a current or next-generation cellular telephone transceiver, a wireless local loop (WLL) transceiver, etc. A Internet connection may already be available on the local network, in which case only a local network connection may be required.

The printer controller optionally includes a local network connection, to allow the printer to be used from a directly-connected workstation or over a local-area network. Possibilities include 10Base-T and 100Base-T Ethernet, USB and USB 2.0, IEEE 1394 (Firewire), and various emerging home networking standards.

The radio transceiver communicates in the unlicensed 900MHz band normally used by cordless telephones, and uses frequency hopping and collision detection to provide interference-free communication.

9.1 DETAILED DOCUMENT DATA FLOW

The main processor receives and verifies the document's page layouts and page objects by Internet pointcast and multicast. It then runs the appropriate RIP software on the DSPs.

The DSPs rasterize each page description and compress the rasterized page image. The main processor stores each compressed page image in memory. The simplest way to load-balance multiple DSPs is to let each DSP rasterize a separate page. The DSPs can always be kept busy since an arbitrary number of rasterized pages can, in general, be stored in memory. This strategy can lead to poor DSP utilization, however, when rasterizing short documents.

Watermark regions in the page description are rasterized to a contone-resolution bi-level bitmap which is losslessly compressed to negligible size and which forms part of the compressed page image.

The infrared (IR) layer of the printed page contains encoded position tags at a density of about 25 per inch. Each tag encodes the page id, tag position, and control bits. Active areas and pressure-sensitive areas in the page description are rasterized to tag-resolution bi-level bitmaps which do not require compression and which form part of the compressed page image.

The main processor passes back-to-back page images to the duplexed print engines. Each print engine stores the compressed page image into its local memory, and starts the page expansion and printing pipeline. Page expansion and printing is pipelined because it is impractical to store a 114MB bi-level CMY+IR page image in memory.

The first stage of the pipeline expands the JPEG-compressed contone CMY layer, expands the Group 4 Fax-compressed bi-level watermark map, and expands the Group 4 Fax-compressed bi-level black layer, all in parallel. The second stage dithers the contone CMY layer using the dither matrix selected by the watermark map, and composites the bi-level black layer over the resulting bi-level CMY layer. Since there is no black ink used in the Netpage Printer, the black layer is composited with each of C, M and Y. In parallel with

this, the tag encoder generates and encodes the bi-level IR tag data. The last stage prints the bi-level CMY+IR data through the Memjet printhead via the printhead interface.

9.2 PRINT ENGINE ARCHITECTURE

The print engine's page expansion and printing pipeline consists of a standard JPEG decoder, a standard Group 4 Fax decoder, a custom halftoner/compositor unit, a custom position tag encoder, and a custom interface to the Memjet printhead. These are described in detail in Section 12.

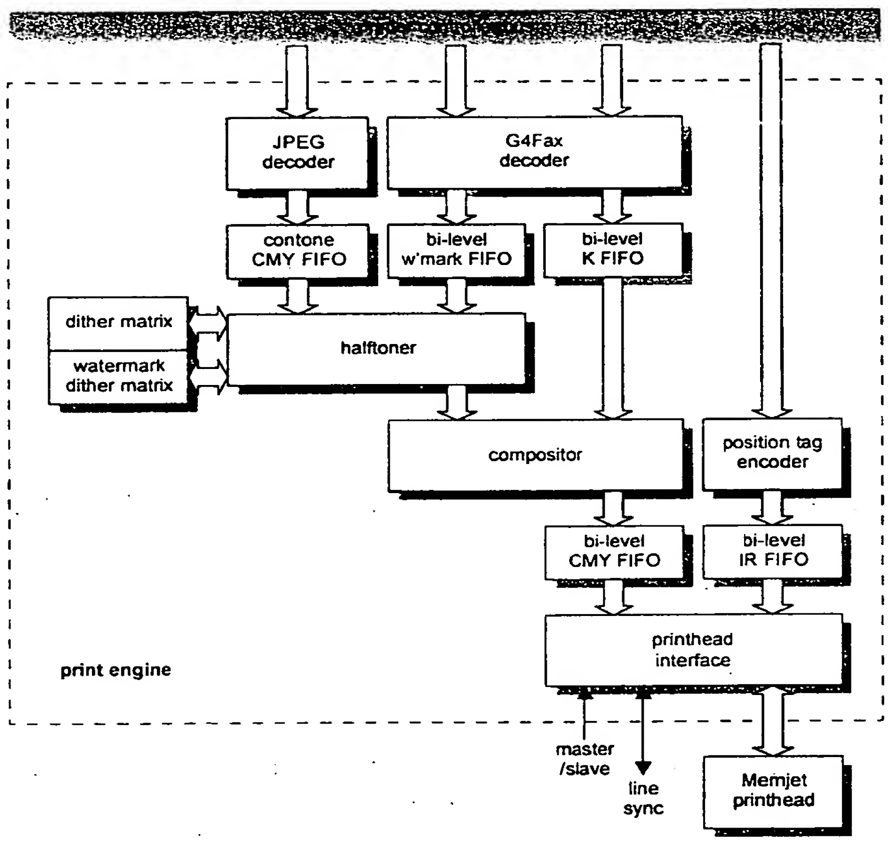


Figure 26. Print engine architecture

When several print engines are used in unison, such as in a duplexed configuration, they are synchronized via a shared line sync signal. Only one print engine, selected via the external master/slave pin, generates the line sync signal onto the shared line.

In the 8½" versions of the Netpage Printer, the two print engines each prints 30 Letter pages per minute along the long dimension of the page (11"), giving a line rate of 8.8 kHz at 1600 dpi. In the 11" Pro versions of the Netpage Printer, the two print engines each prints 45 Letter pages per minute along the short dimension of the page (8½"), giving a line rate of 10.2 kHz. These line rates are well within the operating frequency of the Memjet printhead, which in the current design exceeds 30 kHz.

10 Pen Controller Architecture

The Netpage Pen controller consists of a controlling processor, a radio transceiver, a tilt sensor, a nib pressure sensor, a IR image sensor, flash memory, and 512KB of DRAM, as illustrated in Figure 27.

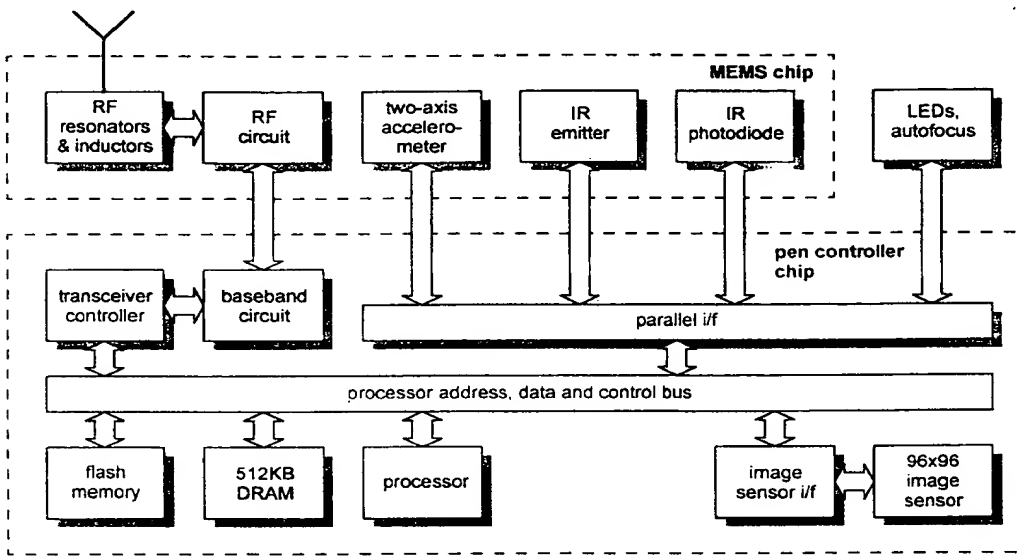


Figure 27. Pen controller architecture

The controlling processor captures and decodes IR position tags from the page via the image sensor, monitors the tilt and pressure sensors, controls the autofocus voice-coil, controls the user interface (tricolor LED), and handles wireless communication with the local Netpage Printer. It consists of a medium-performance (~25MHz RISC) general-purpose microprocessor.

Two-axis tilt sensing is provided by a two-axis accelerometer. Nib pressure sensing is provided by an IR emitter and photodiode pair in conjunction with a reflector coupled with the sprung nib.

The processor, digital transceiver components, 96x96 image sensor, flash memory and 512KB DRAM are integrated in a single ASIC. The MEMS and analog RF components, accelerometers, and the IR emitter/photodiode are integrated in a single MEMS chip, also used in the Netpage Printer.

The radio transceiver communicates in the unlicensed 900MHz band normally used by cordless telephones, and uses frequency hopping and collision detection to provide interference-free communication.

11 Print Engine Page Format

This section describes the format of compressed pages expected by the print engines.

The raster image processors (RIPs) generate pages in this format. The compressed format and the print engines are designed to allow real-time page expansion during printing, to ensure that printing is never interrupted in the middle of a page due to data underrun.

The Netpage Printer reproduces black at full dot resolution (1600 dpi), but reproduces contone color at a somewhat lower resolution using halftoning. The page description is therefore divided into a black layer and a contone layer. The black layer is defined to composite *over* the contone layer.

The black layer consists of a bitmap containing a 1-bit *opacity* for each pixel. This black layer *matte* has a resolution which is an integer factor of the printer's dot resolution. The highest supported resolution is 1600 dpi, i.e. the printer's full dot resolution.

The contone layer consists of a bitmap containing a 24-bit CMY *color* for each pixel. This contone image has a resolution which is an integer factor of the printer's dot resolution. The highest supported resolution is 267 ppi, i.e. one-sixth the printer's dot resolution.

The contone resolution is also typically an integer factor of the black resolution, to simplify calculations in the RIPs. This is not a requirement, however.

The black layer and the contone layer are both in compressed form for efficient storage in the printer's internal memory.

11.1 Page Structure

The Netpage Printer prints with full edge bleed using an 8½" printhead. It imposes no margins and so has a printable page area which corresponds to the size of its paper. The target page size is constrained by the printable page area, less the explicit (target) left and top margins specified in the page description. These relationships are illustrated below.

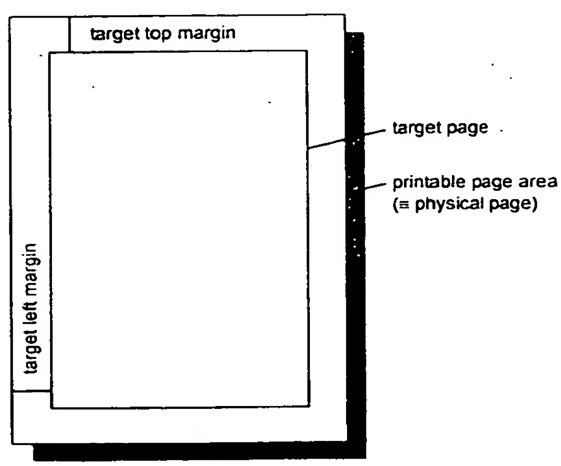


Figure 28. Page structure

11.2 COMPRESSED PAGE FORMAT

Apart from being implicitly defined in relation to the printable page area, each page description is complete and self-contained. There is no data stored separately from the page description to which the page description refers.

The page description consists of a page header which describes the size and resolution of the page, followed by one or more page bands which describe the actual page content.

Table 1 shows the format of the page header.

Table 1. Page header format

field	format	description
signature	16-bit integer	Page header format signature.
version	16-bit integer	Page header format version number.
structure size	16-bit integer	Size of page header.
target resolution (dpi)	16-bit integer	Resolution of target page. This is always 1600 for the Netpage Printer.
target page width	16-bit integer	Width of target page, in dots.
target page height	16-bit integer	Height of target page, in dots.
target left margin	16-bit integer	Width of target left margin, in dots.
target top margin	16-bit integer	Height of target top margin, in dots.
black scale factor	16-bit integer	Scale factor from black resolution to target resolution (must be 2 or greater).
black page width	16-bit integer	Width of black page, in black pixels.
black page height	16-bit integer	Height of black page, in black pixels.
contone scale factor	16-bit integer	Scale factor from contone resolution to target resolution (must be 6 or greater).
contone page width	16-bit integer	Width of contone page, in contone pixels.
contone page height	16-bit integer	Height of contone page, in contone pixels.

The page header contains a signature and version which allow the print engine to identify the page header format. If the signature and/or version are missing or incompatible with the print engine, then the print engine can reject the page.

The page header defines the resolution and size of the target page. The black and contone layers are clipped to the target page if necessary. This happens whenever the black or contone scale factors are not factors of the target page width or height.

The target left and top margins define the positioning of the target page within the printable page area.

The black layer parameters define the pixel size of the black layer, and its integer scale factor to the target resolution.

The contone layer parameters define the pixel size of the contone layer, and its integer scale factor to the target resolution.

Table 2 shows the format of the page band header.

Table 2. Page band header format

signature	16-bit integer	Page band header format signature.
version	16-bit integer	Page band header format version number.
structure size	16-bit integer	Size of page band header.
black band height	16-bit integer	Height of black band, in black pixels.
black band data size	32-bit integer	Size of black band data, in bytes.
contone band height	16-bit integer	Height of contone band, in contone pixels.
contone band data size	32-bit integer	Size of contone band data, in bytes.
watermark map band data size	32-bit integer	Size of watermark map band data, in bytes.
tag control band height	16-bit integer	Height of position tag control band, in tags.
active area map band size	32-bit integer	Size of active area map band data, in bytes.
pressure-sensitive area map band size	32-bit integer	Size of pressure-sensitive area band data, in bytes.

The black layer parameters define the height of the black band, and the size of its compressed band data. The variable-size black data follows the page band header.

The contone layer parameters define the height of the contone band, and the size of its compressed page data, consisting of the contone color data and the associated bi-level watermark map. The variable-size contone data follows the black data. The variable-size bi-level watermark map data follows the contone data.

The tag layer parameters define the height of the position tag control band, and the size of its active area and pressure-sensitive area map band data. The variable-size active area map data follows the watermark map data. The variable-size pressure-sensitive area map data follows the active area map data.

Table 3 shows the format of the variable-size compressed band data which follows the page band header.

Table 3. Page band data format

field	format	description
black data	G4Fax bytestream	Compressed bi-level black data.
contone data	JPEG bytestream	Compressed contone CMY data.
watermark map	G4Fax bytestream	Compressed bi-level watermark map data.
active area map	bitmap	Bi-level active area map data.
pressure-sensitive area map	bitmap	Bi-level pressure-sensitive area map data.

Each variable-size segment of band data is aligned to an 8-byte boundary.

The following sections describe the format of the compressed bi-level layers and the compressed contone layer.

11.2.1 Bi-level Data Compression

11.2.1.1 Group 3 and 4 Facsimile Compression

The Group 3 Facsimile compression algorithm [12] losslessly compresses bi-level data for transmission over slow and noisy telephone lines. The bi-level data represents scanned black text and graphics on a white background, and the algorithm is tuned for this class of images (it is explicitly not tuned, for example, for *halftoned* bi-level images). The *1D* Group 3 algorithm runlength-encodes each scanline and then Huffman-encodes the resulting runlengths. Runlengths in the range 0 to 63 are coded with *terminating* codes. Runlengths in the range 64 to 2623 are coded with *make-up* codes, each representing a multiple of 64, followed by a terminating code. Runlengths exceeding 2623 are coded with multiple make-up codes followed by a terminating code. The Huffman tables are fixed, but are separately tuned for black and white runs (except for make-up codes above 1728, which are common). When possible, the *2D* Group 3 algorithm encodes a scanline as a set of short edge deltas (0, ±1, ±2, ±3) with reference to the previous scanline. The delta symbols are entropy-encoded (so that the zero delta symbol is only one bit long etc.) Edges within a 2D-encoded line which can't be delta-encoded are runlength-encoded, and are identified by a prefix. 1D- and 2D-encoded lines are marked differently. 1D-encoded lines are generated at regular intervals, whether actually required or not, to ensure that the decoder can recover from line noise with minimal image degradation. 2D Group 3 achieves compression ratios of up to 6:1 [12].

The Group 4 Facsimile algorithm [6] losslessly compresses bi-level data for transmission over *error-free* communications lines (i.e. the lines are truly error-free, or error-correction is done at a lower protocol level). The Group 4 algorithm is based on the 2D Group 3 algorithm, with the essential modification that since transmission is assumed to be error-free, 1D-encoded lines are no longer generated at regular intervals as an aid to error-recovery. Group 4 achieves compression ratios ranging from 20:1 to 60:1 for the CCITT set of test images [12].

The design goals and performance of the Group 4 compression algorithm qualify it as a compression algorithm for the bi-level layers. However, its Huffman tables are tuned to a lower scanning resolution (100-400 dpi), and it encodes runlengths exceeding 2623 awkwardly. At 800 dpi, our maximum runlength is currently 6400. Although a Group 4 decoder core might be available for use in the printer controller chip, it might not handle runlengths exceeding those normally encountered in 400 dpi facsimile applications, and so would require modification.

11.2.1.2 Edge Delta / Runlength (EDRL) Compression

Since most of the benefit of Group 4 comes from the delta-encoding, a simpler algorithm based on delta-encoding alone is likely to meet our requirements. This approach, in the form of the Edge Delta / Runlength (EDRL) encoding, is described in detail in another Silverbrook design document [3]. There are therefore two viable approaches to losslessly compressing bi-level data: Group 4 Facsimile and EDRL. Their compression performance is compared below. Where this document refers to Group 4 Facsimile (G4Fax) compression, EDRL can be substituted, purely as an implemetation decision.

Magazine text is typically typeset in a typeface with serifs (such as Times) at a point size of 10. At this size a Letter page holds up to 14,000 characters, though a typical magazine page holds only about 7,000 characters. Text is seldom typeset at a point size smaller than 5. At 800 dpi, text cannot be meaningfully rendered at a point size lower than 2 using a standard typeface. Table 4 illustrates the legibility of various point sizes.

Table 4. Text at different point sizes

[illegible]	[illegible]
2	[illegible]
3	The quick brown fox jumps over the lazy dog.
4	The quick brown fox jumps over the lazy dog.
5	The quick brown fox jumps over the lazy dog.
6	The quick brown fox jumps over the lazy dog.
7	The quick brown fox jumps over the lazy dog.
8	The quick brown fox jumps over the lazy dog.
9	The quick brown fox jumps over the lazy dog.
10	The quick brown fox jumps over the lazy dog.

Table 5 shows Group 4 and EDRL compression performance on pages of text of varying point sizes, rendered at 800 dpi. The distribution of characters on the test pages is based on English-language statistics [11].

Table 5. Group 4 and EDRL compression performance on text at 800 dpi

point size	characters per page	Group 4 compression ratio	EDRL compression ratio
2	340,000	2.3	1.7
3	170,000	3.2	2.5
4	86,000	4.7	3.8
5	59,000	5.5	4.9
6	41,000	6.5	6.1
7	28,000	7.7	7.4
8	21,000	9.1	9.0
9	17,000	10.2	10.4
10	14,000	10.9	11.3
11	12,000	11.5	12.4
12	8,900	13.5	14.8
13	8,200	13.5	15.0
14	7,000	14.6	16.6
15	5,800	16.1	18.5
20	3,400	19.8	23.9

For a point size of 9 or greater, EDRL slightly outperforms Group 4, simply because Group 4's runlength codes are tuned to 400 dpi.

These compression results bear out the observation that entropy-encoded runlengths contribute much less to compression than 2D encoding, unless the data is poorly correlated vertically, such as in the case of very small characters.

11.2.1.3 Black Layer Compression Format

The 800 dpi black layer is losslessly compressed using G4Fax (or EDRL) at a typical compression ratio exceeding 10:1.

11.2.1.4 Watermark Map Compression Format

The 267 dpi watermark layer, which matches the 267 ppi contone color layer, is losslessly compressed using G4Fax (or EDRL) at a typical compression ratio exceeding 50:1.

11.2.2 Contone Data Compression

11.2.2.1 JPEG Compression

The JPEG compression algorithm [9] lossily compresses a contone image at a specified quality level. It introduces imperceptible image degradation at compression ratios below 5:1, and negligible image degradation at compression ratios below 10:1 [13].

JPEG typically first transforms the image into a color space which separates luminance and chrominance into separate color channels. This allows the chrominance channels to be subsampled without appreciable loss because of the human visual system's relatively greater sensitivity to luminance than chrominance. After this first step, each color channel is compressed separately.

The image is divided into 8×8 pixel blocks. Each block is then transformed into the frequency domain via a discrete cosine transform (DCT). This transformation has the effect of concentrating image energy in relatively lower-frequency coefficients, which allows higher-frequency coefficients to be more crudely quantized. This quantization is the principal source of compression in JPEG. Further compression is achieved by ordering coefficients by frequency to maximize the likelihood of adjacent zero coefficients, and then runlength-encoding runs of zeroes. Finally, the runlengths and non-zero frequency coefficients are entropy coded. Decompression is the inverse process of compression.

11.2.2.2 CMY Contone Layer Compression Format

The CMY contone layer is compressed to an interleaved color JPEG bytestream. The interleaving is required for space-efficient decompression in the printer, but may restrict the decoder to two sets of Huffman tables rather than four (i.e. one per color channel) [13]. If luminance and chrominance are separated, then the luminance channels can share one set of tables, and the chrominance channels the other set..

If luminance/chrominance separation is deemed necessary, either for the purposes of table sharing or for chrominance subsampling, then CMY is converted to YCrCb and Cr and Cb are duly subsampled.

The JPEG bytestream is complete and self-contained. It contains all data required for decompression, including quantization and Huffman tables.

12 Print Engine

12.1 HALFTONER/COMPOSITOR

The halftoner/compositor unit (HCU) (Figure 30) combines the functions of halftoning the contone CMY layer to bi-level CMY, and compositing the black layer over the halftoned contone layer. It also selects between the normal and the "watermark" dither matrix on a pixel-by-pixel basis, based on the corresponding value in the watermark map.

The input to the HCU is an expanded 267 ppi CMY contone layer, an expanded 267 dpi watermark map, and an expanded 1600 dpi black layer. The output from the HCU is a set of 1600 dpi bi-level CMY image lines.

Once started, the HCU proceeds until it detects an *end-of-page* condition, or until it is explicitly stopped via its control register.

The HCU generates a page of dots of a specified width and length. The width and length must be written to the *page width* and *page length* registers prior to starting the HCU. The page width corresponds to the width of the printhead. The page length corresponds to the length of the target page.

The HCU generates target page data between specified left and right margins relative to the page width. The positions of the left and right margins must be written to the *left margin* and *right margin* registers prior to starting the HCU. The distance from the left margin to the right margin corresponds to the target page width.

The HCU consumes black and contone/watermark data according to specified black and contone page widths. These page widths must be written to the *black page width* and *contone page width* registers prior to starting the HCU. The HCU clips black and contone data to the target page width. This allows the black and contone page widths to exceed the target page width without requiring any special end-of-line logic at the input FIFO level.

The relationships between the page width, the black and contone page widths, and the margins are illustrated in Figure 29.

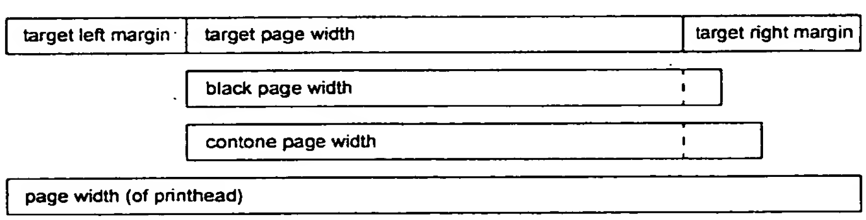


Figure 29. Relationships between page widths and margins

The HCU scales contone data to printer resolution both horizontally and vertically based on a specified scale factor. This scale factor must be written to the *contone scale factor* register prior to starting the HCU.

Table 6. Halftoner/compositor control and configuration registers

start	1	Start the HCU.
stop	1	Stop the HCU.
page width	14	Page width of printed page, in dots. This is the number of dots which have to be generated for each line.
left margin	14	Position of left margin, in dots.
right margin	14	Position of right margin, in dots.
page length	15	Page length of printed page, in dots. This is the number of lines which have to be generated for each page.
black page width	14	Page width of black layer, in dots. Used to detect the end of a black line.
contone page width	14	Page width of contone layer, in dots. Used to detect the end of a contone line.
contone scale factor	4	Scale factor used to scale contone data to bi-level resolution.

The consumer of the data produced by the HCU is the printhead interface. The printhead interface requires bi-level CMY image data in *planar* format, i.e. with the color planes separated. Further, it also requires that even and odd pixels are separated. The output stage of the HCU therefore uses 6 parallel pixel FIFOs, one each for *even cyan*, *odd cyan*, *even magenta*, *odd magenta*, *even yellow*, and *odd yellow*.

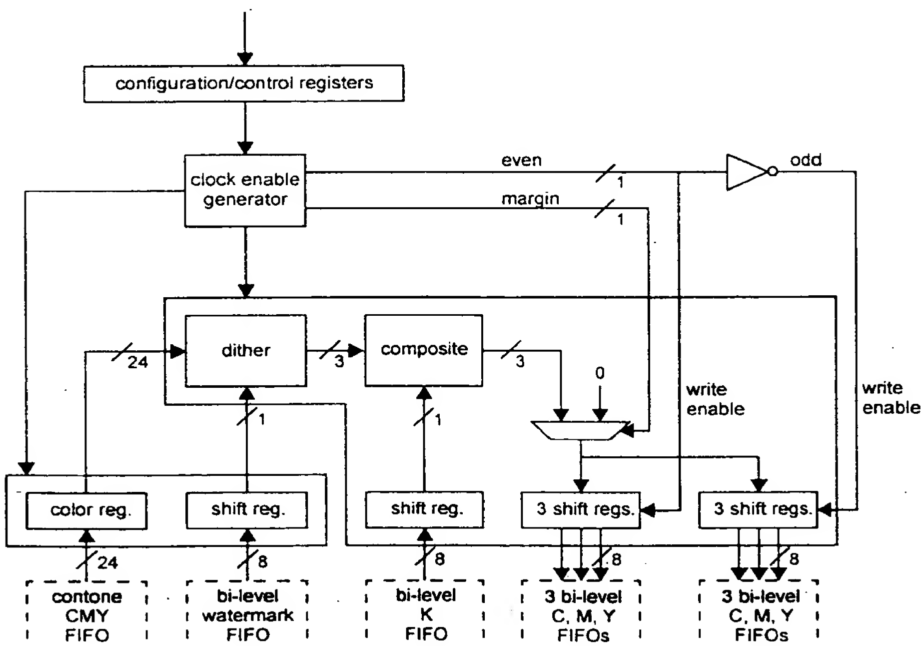


Figure 30. Halftoner/compositor unit

The input contone CMY FIFO is a full 7KB line buffer. The line is used *contone scale factor* times to effect vertical up-scaling via line replication. FIFO write address wrapping is disabled until the start of the last use of the line.

12.1.1 Dither

The dither unit converts a 24-bit contone CMY value into a 3-bit bi-level CMY value. Each color component is independently compared with the 8-bit threshold value at the current dot position in the dither cell (see Figure 31).

There are two programmable 64×64×8-bit dither cells: a normal dither cell and a "watermark" dither cell. The 1-bit watermark value associated with the current contone pixel controls the selection of the watermark or normal threshold value.

The dither cell address generator contains a pair of 6-bit row and column address registers. Both registers are initialized to zero. The row address is incremented on every dot line advance, modulo 64. The column address is incremented on every dot clock, modulo 64, and is reset on a dot line advance.

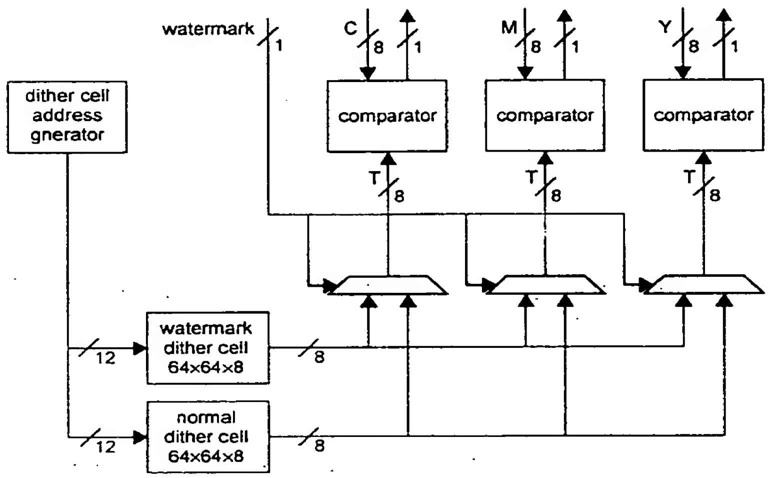


Figure 31. Dither, with watermarking

12.1.2 Composite

The composite unit composites a black layer dot over a halftoned CMY layer dot. If the black layer opacity is one, then the halftoned CMY is set to black (i.e. all ones). The black layer opacity is therefore simply ORed with each of the halftoned C, M, and Y values.

12.1.3 Clock Enable Generator

The clock enable generator generates enable signals for clocking the contone CMY/watermark pixel input, the black dot input, and the CMY dot output.

As described earlier, the contone pixel input buffer is used as both a line buffer and a FIFO. Each line is read once and then used *contone scale factor* times. FIFO write address wrapping is disabled until the start of the final replicated use of the line, at which time the clock enable generator generates a *contone line advance enable* signal which enables wrapping.

The clock enable generator also generates an *even* signal which is used to select the even or odd set of output dot FIFOs, and a *margin* signal which is used to generate white dots when the current dot position is in the left or right margin of the page.

The clock enable generator uses a set of counters. The internal logic of the counters is defined in Table 7. The logic of the clock enable signals is defined in Table 8.

Table 7. Clock enable generator counter logic

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
dot	D	14	page width	RP[a] ∨ EOL[b]	(D>0) ∧ clk
line	L	15	page length	RP	(L>0) ∧ EOL
left margin	LM	14	left margin	RP ∨ EOL	(LM>0) ∧ clk
right margin	RM	14	right margin	RP ∨ EOL	(RM>0) ∧ clk
even/odd dot	E	1	0	RP ∨ EOL	clk
black dot	BD	14	black width	RP ∨ EOL	(LM=0) ∧ (BD>0) ∧ clk
contone dot	CD	14	contone width	RP ∨ EOL	(LM=0) ∧ (CD>0) ∧ clk
contone sub-pixel	CSP	4	contone scale factor	RP ∨ EOL ∨ (CSP=0)	(LM=0) ∧ clk
contone sub-line	CSL	4	contone scale factor	RP ∨ (CSL=0)	EOL ∧ clk

a. RP (reset page) condition: external signal
b. EOL (end-of-line) condition: (D=0) ∧ (BD=0) ∧ (CD=0)

Table 8. Clock enable generator output signal logic

output signal	condition
output dot clock enable	(D>0) ∧ ¬EOP[a]
black dot clock enable	(LM=0) ∧ (BD>0) ∧ ¬EOP
contone pixel clock enable	(LM=0) ∧ (CD>0) ∧ (CSP=0) ∧ ¬EOP
contone line advance enable	(CSL=0) ∧ ¬EOP
even	E=0
margin	(LM=0) ∨ (RM=0)

a. EOP (end-of-page) condition: L=0

12.2 PRINTHEAD INTERFACE

Netpage Printer uses an 8½" CMYK Memjet printhead, as described in Section 14. The printhead consists of 17 segments arranged in 2 segment groups. The first segment group contains 9 segments, and the second group contains 8 segments. There are 13,600 nozzles of each color in the printhead, making a total of 54,400 nozzles.

The printhead interface is a standard Memjet printhead interface, as described in Section 15, configured with the following operating parameters:

- MaxColors = 4
- SegmentsPerXfer = 9
- SegmentGroups = 2

Although the printhead interface has a number of external connections, not all used for an 8½" printhead, so not all are connected to external pins on the print engine. Specifically, the value for SegmentGroups implies that there are only 2 SRClock pins and 2 SenseSegSelect pins. All 36 ColorData pins are required, however.

12.3 Position Tag Encoder

The position tag encoder encodes the page id of the page being printed, together with the current x-y position on the page, into an error-correctably encoded position tag which is subsequently printed in infrared (IR) ink on the page.

The position tag encoder takes the following as input:

- the page id
- a pressure sensitive area bitmap at tag resolution
- an active area bitmap at tag resolution

It writes a bi-level IR bitstream to the bi-level IR FIFO. The position tag encoder design as defined here allows a single page id per printed page, and provides for a page size of up to 21.7 inches per dimension ($17 \times 4 \times 512 / 1600$). In addition, the position tag encoder allows for both landscape and portrait orientations.

The position tag encoder consists of two stages connected by a buffer. The first stage, the *tag data encoder*, Reed-Solomon encodes the data to be placed in the tags. The second stage, the *tag formatter*, places the encoded data into the tag format and passes it on to the bi-level IR FIFO. Since the tag formatter runs one line of tags behind the tag data encoder, the tag data buffer is in fact a double-buffer. The relationship is shown in Figure 32.

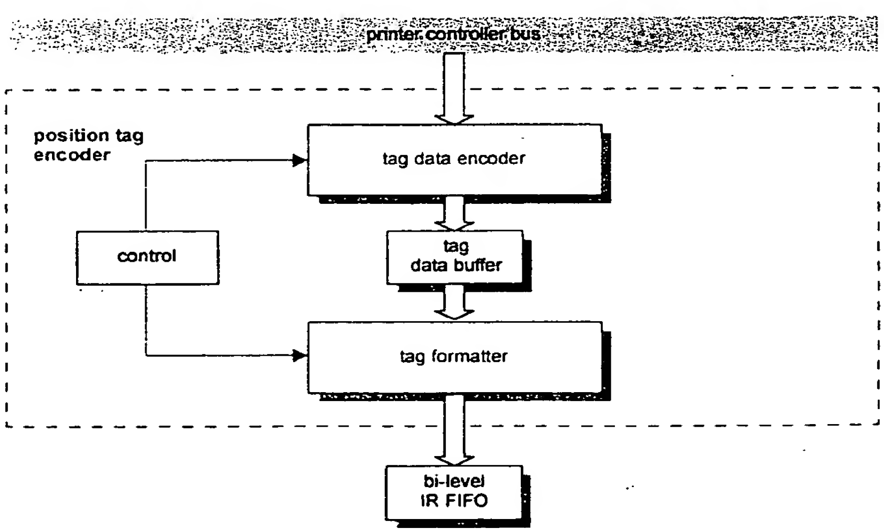


Figure 32. Two stages of position tag encoder

The position tag encoder provides full macrodot-to-dot conversion, thereby eliminating any special knowledge of macrodots from the bi-level FIFO user. However, the position tag encoder itself does make use of the redundancy caused by dot replication in macrodot

formation with respect to the dot axis only. Since the macrodot-to-dot ratio is 1:4, the position tag encoder operates at 1/4 of the dot frequency (60 MHz).

12.3.1 Tag Data Buffer

The tag data buffer consists of two identical buffers. One is read by the tag formatter while the other is being written to by the tag data encoder. The two buffers are then logically swapped.

Each buffer contains $256T$ bits, where T is the maximum number of tags in a given printed line of dots. The maximum amount of memory required for each buffer is 16KBytes (when $T = 512$). To support a 12-inch printhead, $T = 283$, so the memory required for each buffer is 9056 bytes (55% of the maximum). Note that the remainder of the position tag encoder does not change in size based on T.

As the encoded bits are generated by the tag data encoder, they are written to consecutive bits in the tag data buffer. The data bits are read out in random-access fashion by the tag formatter. Having 256 bits allocated to each tag allows the tag number to be used for generating the high bits of the address and the low 8 bits of the address to come from the sub-tag address generator.

The tag data buffer therefore has the structure as shown by Figure 33:

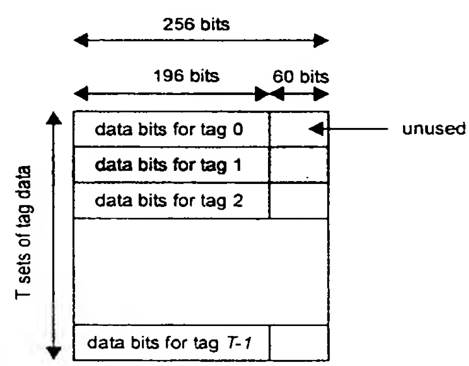


Figure 33. Tag data buffer logical structure

12.3.2 Tag Data Encoder

It is the responsibility of the tag data encoder to produce the 28 bits of encoded mode data and 168 bits of encoded tag data for a total of 196 bits of encoded data. The encoding scheme for position tags is defined in [5].

Each tag is represented on the page by a 17×17×4×4 structure of dots. However, since the tag data formatter is running at only 1/4 of the dot frequency, this results in a time of 1156 cycles (17×17×4) per tag.

The tag data encoder and control block functions are provided by a simple microprocessor core running at 60 MHz. This is desirable compared to the design effort to implement a specific Reed-Solomon encoder for two types of encoding and specific tag codeword construction.

The 196 bits of encoded tag data are written to one of the tag data buffers while the other is being read from by the tag formatter function.

The coordinates encoded within the tags for a given line of tags will depend on the width of the printhead and whether or not the pages are being printed in portrait or landscape mode, as shown in Figure 34.

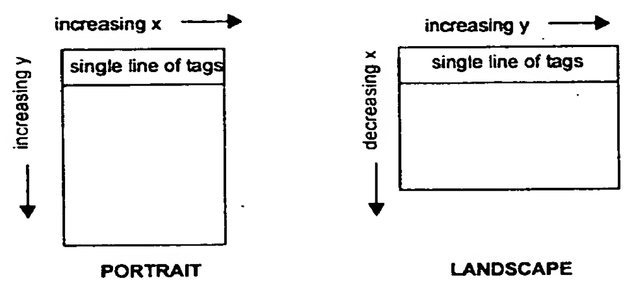


Figure 34. Relationship of coordinates to tags in a tag line

- If processing a *portrait* page, each printed line corresponds to changes in x coordinates. A printed line of tags therefore has a constant y coordinate, a start x coordinate of *0*, and end x coordinate of *T-1* with a Δx of *1*. To advance to the next line of tags, the y coordinate increases by 1 and x is reset to 0.
- If processing a *landscape* page, each printed line corresponds to changes in y coordinates. A printed line of tags therefore has a constant x coordinate, a start y coordinate of 0, and an end y coordinate of *T-1* with a Δy of *1*. To advance to the next line of tags, the x coordinate is decreased by 1 and y is reset to 0.

The high level control function for coordinate generation is outlined here in pseudocode.

```
if portraitMode
   For tagline = 0 to maxTagLines
      For tag = 0 to lastTagInLine
         EncodeTag(tag, pageId, tag, tagLine)
      EndFor
   EndFor
else
   tagX = maxTagLines
   For tagline = 0 to maxTagLines
      For tag = 0 to lastTagInLine
         EncodeTag(tag, pageId, tagX, tag)
         tagX = tagX - 1
      EndFor
   EndFor
EndIf
```

In relation to the EncodeTag function, an examination of the tag structure reveals:

- of the 168 data bits, 90 bits are copied to a maximum of 108 bits in the output with only trivial change. The remaining bits are generated check words based on GF(64).

- of the 28 mode bits, 8 bits are directly copied to the output. The remaining 20 are generated check words based on GF(16).

Therefore the 8 mode bits output and at least 90 bits of the data bits output can be output at a rate approximating 1 cycle per bit. The remaining 70 bits must be generated using Reed-Solomon encoding. This allows approximately 1000 cycles for 70 bits, or approximately 14 cycles per encoded bit. Encoding over GF(16) and GF(64) can be accomplished using small tables totalling 64 and 256 bytes respectively.

The bits are stored in the tag data buffer in bit generation order in order to simplify the writing process. The reading process (in the tag formatter) has random access to the generated bits, but has specific addressing hardware to assist in this task. The order for writing bits is as follows:

- 90-108 bits of data (15-18 × 6-bit codewords, depending on whether 1s or 0s had to be inserted to generate non 000000 and non 111111 codewords) based on the 90 bits of data (64 bits of pageId, 9 bits of tagX, 9 bits of tagY, 1 bit from the active area bitmap at tag resolution, 1 bit from the pressure sensitive area bitmap at tag resolution and 6 reserved bits).
- 78-60 bits of check words (13-10 × 6-bit check words, depending on how many data codewords were written)
- 8 bits of mode data (2 × 4-bit codewords)
- 20 bits of check words (5 × 4-bit check words)

12.3.3 Tag Formatter

The tag formatter is responsible for merging the encoded tag data with the tag structure, and placing the dots in the IR buffer in the correct order for printing. The encoded tag data is read from the tag data buffer as previously generated by the tag data encoder. The formatted dots are placed in the bi-level IR buffer such that the same data is clocked in 4 times. This allows the tag formatter to run at 1/4 of the dot rate.

We use a simple set of counters for formatting a set of 68 lines of dots (a set of 68 lines equates to 17 rows of 4-dot macrodots). The logic for the 68 lines of dots can be repeated until the page has finished printing. The within-tag counters index into an address table of 9-bit entries with the following attributes:

- If the entry's high bit = 0, then the macrodot comes from an encoded data bit. The encoded data bit to use is given by the concatenation of the current tag# (10 bits) and 8 bits of address from the low 8-bits of the entry.
- If the entry's high bit = 1, then the macrodot comes from the constant structure of a position tag (the bull's-eye and orientation bits). The entry's low bit determines the state of the macrodot.

The address table defines the structure of a positional tag at the macrodot level. It therefore has a constant size of 289 × 9-bit entries. The stored 8-bit addresses simply combine the tag encoding structure with the order that the bits are written to the tag data buffer. For the constant part of the tag (the bull's-eye and orientation bits), the entries are simply 100000000 for a white dot and 100000001 for a black dot.

The tag formatter is shown in pseudocode form:

```
For line = 0 to 16
```

```
    For dotY = 0 to 3
       For tag = 0 to numTagsInLine
          For pixel = 0 to 16
             adr = TableLookup[line, pixel]
             If (adr5=0)
                bit = TagDataBuffer[tag | adr0-4]
             Else
                bit = adr0
             EndIf
             Place 4 copies of bit in FIFO
          EndFor
       EndFor
    EndFor
EndFor
Swap TagDataBuffers
```

The pseudocode is readily transferred to logic, as illustrated by Figure 35.

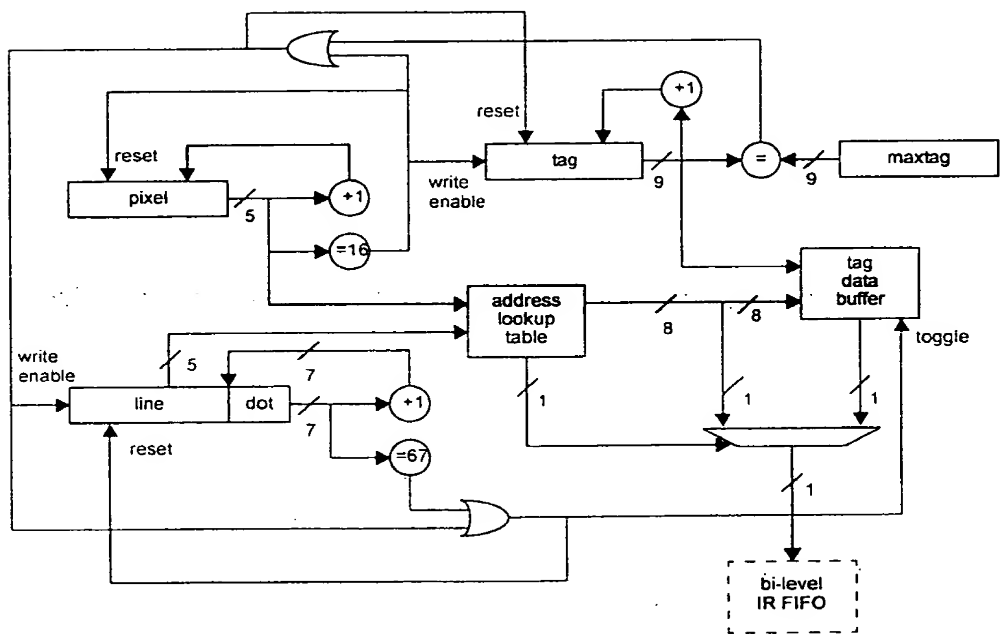


Figure 35. Tag formatter

13 Print Engine Driver

This section discusses rasterizing pages to the internal compressed page format expected by the print engine, as defined in Section 11, in terms of the compressed page driver which hides this device-dependent behavior from the higher-level raster image processor (RIP). The relationship between the RIP, the graphics system, the compressed page driver, and the print engine, is illustrated in Figure 36.

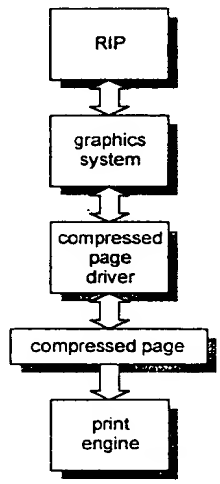


Figure 36. RIP and print engine driver

The RIP and the graphics system are considered generic components and are not defined further at this stage.

13.1 Graphics and Imaging Model

The compressed page driver is closely coupled with the graphics system used by the RIP, so that the driver can provide device-specific handling for different graphics and imaging operations, in particular compositing operations and text operations.

The graphics system renders images and graphics to a nominal resolution specified by the compressed page driver, but allows the compressed page driver to take control of rendering text. In particular, the graphics system provides sufficient information to the compressed page driver to allow it to render and position text at a higher resolution than the nominal device resolution.

The host graphics system requires random access to a contone page buffer at the nominal device resolution, into which it composites graphics and imaging objects, but it allows the compressed page driver to take control of the actual compositing - i.e. the compressed page driver manages the page buffer.

13.2 Two-Layer Page Buffer

The compressed page format contains a 267 ppi contone layer and an 800 dpi black layer. The black layer is conceptually *above* the contone layer, i.e. the black layer is composited *over* the contone layer by the print engine. The compressed page driver therefore main-

tains a page buffer which correspondingly contains a medium-resolution contone layer and a high-resolution black layer.

The graphics systems renders and composites objects into the page buffer bottom-up - i.e. later objects obscure earlier objects. This works naturally when there is only a single layer, but not when there are two layers which will be composited later. It is therefore necessary to detect when an object being placed on the contone layer obscures something on the black layer.

When obscuration is detected, the obscured black pixels are composited with the contone layer and removed from the black layer. The obscuring object is then laid down on the contone layer, possibly interacting with the black pixels in some way. If the compositing mode of the obscuring object is such that no interaction with the background is possible, then the black pixels can simply be discarded without being composited with the contone layer. In practice, of course, there is little interaction between the contone layer and the black layer, since images and text rarely overlap.

The compressed page driver specifies a nominal page resolution of 267 ppi to the graphics system. Where possible the compressed page driver relies on the graphics system to render image and graphics objects to the pixel level at 267 ppi, with the exception of *black* text. The compressed page driver fields all text rendering requests, detects and renders black text at 800 dpi, but returns non-black text rendering requests to the graphics system for rendering at 267 ppi.

Ideally the graphics system and the compressed page driver manipulate color in device-independent RGB, deferring conversion to device-specific CMY until the page is complete and ready to be sent to the printer. This reduces page buffer requirements and makes compositing more rational. Compositing in in a device-dependent color space is not ideal.

Ultimately the graphics system asks the compressed page driver to composite each rendered object into the compressed page driver's page buffer. Each such object uses 24-bit contone RGB, and has an explicit (or implicitly opaque) opacity channel.

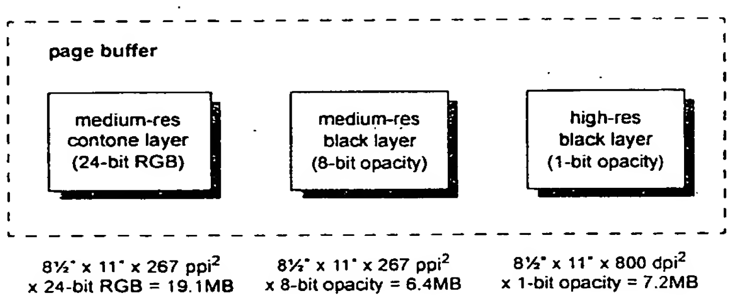


Figure 37. Two-layer page buffer

The compressed page driver maintains the two-layer page buffer in three parts. The first part is the medium-resolution (267 ppi) contone layer. This consists of a 24-bit RGB bitmap. The second part is a medium-resolution black layer. This consists of an 8-bit opacity bitmap. The third part is a high-resolution (800 dpi) black layer. This consists of a 1-bit opacity bitmap. The medium-resolution black layer is a subsampled version of the high-resolution opacity layer. In practice, assuming the medium resolution is an integer factor n of the high resolution (e.g. $n = 800 / 267 = 3$), each medium-resolution opacity

value is obtained by averaging the corresponding $n \times n$ high-resolution opacity values. This corresponds to box-filtered subsampling. The subsampling of the black pixels effectively antialiases edges in the high-resolution black layer, thereby reducing ringing artifacts when the contone layer is subsequently JPEG-compressed and decompressed.

The structure and size of the page buffer is illustrated in Figure 37.

13.3 COMPOSITING MODEL

For the purposes of discussing the page buffer compositing model, we define the following variables.

Table 9. Compositing variables

variable	description	resolution	format
n	medium to high resolution scale factor	-	-
C_C	contone layer color	medium	8-bit color component
C_G	contone object color	medium	8-bit color component
α_G	contone object opacity	medium	8-bit opacity
α_L	medium-resolution black layer opacity	medium	8-bit opacity
α_B	black layer opacity	high	1-bit opacity
α_T	black object opacity	high	1-bit opacity

When a black object of opacity α_T is composited with the black layer, the black layer is updated as follows:

$$\alpha_{B_{x,y}} = \alpha_{B_{x,y}} \vee \alpha_{T_{x,y}} \qquad (1)$$

$$\alpha_{L_{x,y}} = \frac{1}{n^2} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} 255 \alpha_{B_{nx+i,\, ny+j}} \qquad (2)$$

The object opacity is simply *or*ed with the black layer opacity (Eq. 1), and the corresponding part of the medium-resolution black layer is re-computed from the high-resolution black layer (Eq. 2).

When a contone object of color C_G and opacity α_G is composited with the contone layer, the contone layer and the black layer are updated as follows:

$$C_{C_{x,y}} = C_{C_{x,y}}(1 - \alpha_{L_{x,y}}) \;\text{ if }\; (\alpha_{G_{x,y}} > 0) \qquad (3)$$

$$\alpha_{L_{x,y}} = 0 \;\text{ if }\; (\alpha_{G_{x,y}} > 0) \qquad (4)$$

$$\alpha_{B_{x,y}} = 0 \;\text{ if }\; (\alpha_{G_{x/n,\, y/n}} > 0) \qquad (5)$$

$$C_{C_{x,y}} = C_{C_{x,y}}(1 - \alpha_{G_{x,y}}) + C_{G_{x,y}} \alpha_{G_{x,y}} \qquad (6)$$

Wherever the contone object hides the black layer, even if not fully opaquely, the affected black layer pixels are composited with the contone layer (Eq. 3) and removed from the

black layer (Eq. 4 and Eq. 5). The contone object is then composited with the contone layer (Eq. 6).

13.4 PAGE COMPRESSION

Once page rendering is complete, the compressed page driver converts the contone layer to printer-specific CMY with the help of color management functions provided by the graphics system.

The compressed page driver then compresses and packages the black layer and the contone layer into the compressed page format defined in Section 11. This is subsequently expanded in real time by the print engine.

The forward discrete cosine transform (DCT) is the costliest part of JPEG compression. In current high-quality software implementations, the forward DCT of each 8x8 block requires 12 integer multiplications and 32 integer additions [10]. This places a trivial additional load on each RIP DSP.

MEMJET PRINTHEAD

14 Memjet Printhead

A Memjet printhead is a drop-on-demand 1600 dpi inkjet printer that produces bi-level dots in up to 4 colors to produce a printed page of a particular width. Since the printhead prints dots at 1600 dpi, each dot is approximately 22.5μm in diameter, and spaced 15.875μm apart. Because the printing is bi-level, the input image should be dithered or error-diffused for best results.

Typically a Memjet printhead for a particular application is page-width. This enables the printhead to be stationary and allows the paper to move past the printhead. Figure 38 illustrates a typical configuration.

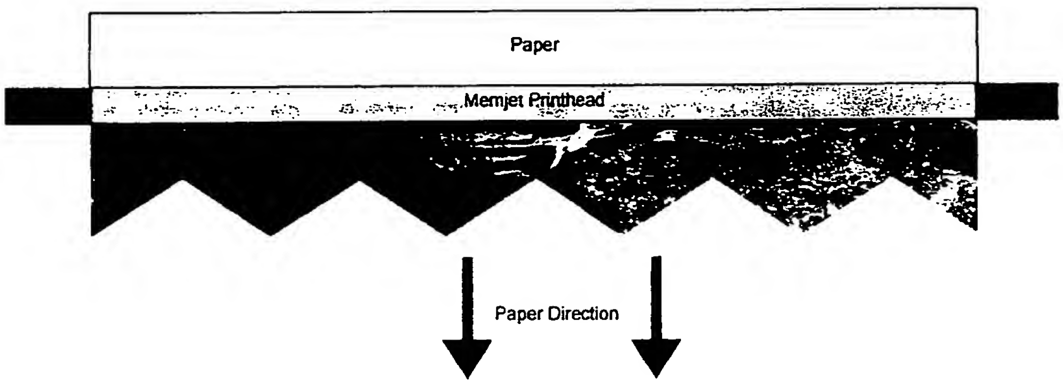


Figure 38. A Memjet printhead

A Memjet printhead is composed of a number of identical 1/2 inch Memjet segments. The segment is therefore the basic building block for constructing a printhead.

14.1 The Structure of a Memjet Segment

This section examines the structure of a single segment, the basic building block for constructing Memjet printheads.

14.1.1 Grouping of Nozzles Within a Segment

The nozzles within a single segment are grouped for reasons of physical stability as well as minimization of power consumption during printing. In terms of physical stability, a total of 10 nozzles share the same ink reservoir. In terms of power consumption, groupings are made to enable a low-speed and a high-speed printing mode.

Memjet segments support two printing speeds to allow speed/power consumption trade-offs to be made in different product configurations.

In the low-speed printing mode, 4 nozzles of each color are fired from the segment at a time. The exact number of nozzles fired depends on how many colors are present in the printhead. In a four color (e.g. CMYK) printing environment this equates to 16 nozzles firing simultaneously. In a three color (e.g. CMY) printing environment this equates to 12 nozzles firing simultaneously. To fire all the nozzles in a segment, 200 different sets of nozzles must be fired.

In the high-speed printing mode, 8 nozzles of each color are fired from the segment at a time. The exact number of nozzles fired depends on how many colors are present in the printhead. In a four color (e.g. CMYK) printing environment this equates to 32 nozzles firing simultaneously. In a three color (e.g. CMY) printing environment this equates to 24 nozzles firing simultaneously. To fire all the nozzles in a segment, 100 different sets of nozzles must be fired.

The power consumption in the low-speed mode is half that of the high-speed mode. Note, however, that the energy consumed to print a page is the same in both cases.

14.1.1.1 10 Nozzles Make a Pod

A single pod consists of 10 nozzles sharing a common ink reservoir. 5 nozzles are in one row, and 5 are in another. Each nozzle produces dots 22.5μm in diameter spaced on a 15.875μm grid to print at 1600 dpi. Figure 39 shows the arrangement of a single pod, with the nozzles numbered according to the order in which they must be fired.

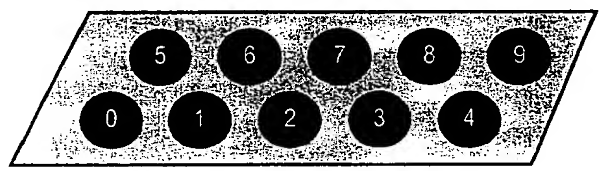


Figure 39. A single pod, numbered by firing order

Although the nozzles are fired in this order, the relationship of nozzles and physical placement of dots on the printed page is different. The nozzles from one row represent the even dots from one line on the page, and the nozzles on the other row represent the odd dots from the adjacent line on the page. Figure 40 shows the same pod with the nozzles numbered according to the order in which they must be loaded.

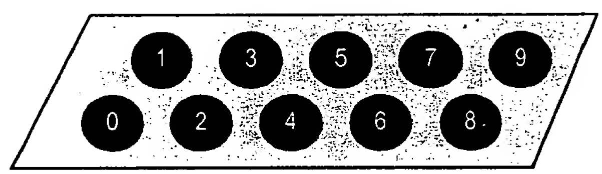


Figure 40. A single pod, numbered by load order

The nozzles within a pod are therefore logically separated by the width of 1 dot. The exact distance between the nozzles will depend on the properties of the Memjet firing mechanism. The printhead is designed with staggered nozzles designed to match the flow of paper.

14.1.1.2 1 Pod of Each Color Makes a Chromapod

One pod of each color are grouped together into a ***chromapod***. The number of pods in a chromapod will depend on the particular application. In a monochrome printing system (such as one that prints only black), there is only a single color and hence a single pod. Photo printing application printheads require 3 colors (cyan, magenta, yellow), so Memjet segments used for these applications will have 3 pods per chromapod (one pod of each color). The expected maximum number of pods in a chromapod is 4, as used in a CMYK (cyan, magenta, yellow, black) printing system (such as a desktop printer). This maximum of 4 colors is not imposed by any physical constraints - it is merely an expected maximum

from the expected applications (of course, as the number of colors increases the cost of the segment increases and the number of these larger segments that can be produced from a single silicon wafer decreases).

A chromapod represents different color components of the same horizontal set of 10 dots on different lines. The exact distance between different color pods depends on the Memjet operating parameters, and may vary from one Memjet design to another. The distance is considered to be a constant number of dot-widths, and must therefore be taken into account when printing: the dots printed by the cyan nozzles will be for different lines than those printed by the magenta, yellow or black nozzles. The printing algorithm must allow for a variable distance up to about 8 dot-widths between colors. Figure 41 illustrates a single chromapod for a CMYK printing application.

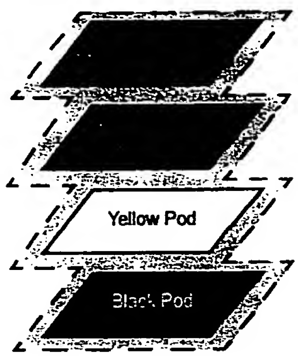


Figure 41. A Single Chromapod Contains 1 Pod of each Color

14.1.1.3 5 Chromapods Make a Podgroup

5 chromapods are organized into a single ***podgroup***. A podgroup therefore contains 50 nozzles for each color. The arrangement is shown in Figure 42, with chromapods numbered 0-4 and using a CMYK chromapod as the example. Note that the distance between adjacent chromapods is exaggerated for clarity.

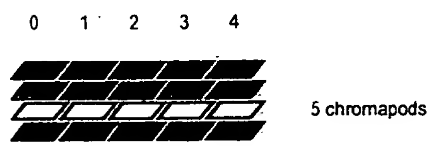


Figure 42. A Single Podgroup Contains 5 Chromapods

14.1.1.4 2 Podgroups Make a Phasegroup

2 podgroups are organized into a single ***phasegroup***. The phasegroup is so named because groups of nozzles within a phasegroup are fired simultaneously during a given firing phase (this is explained in more detail below). The formation of a phasegroup from 2 podgroups is entirely for the purposes of low-speed and high-speed printing via 2 PodgroupEnable lines.

During low-speed printing, only one of the two PodgroupEnable lines is set in a given firing pulse, so only one podgroup of the two fires nozzles. During high-speed printing, both PodgroupEnable lines are set, so both podgroups fire nozzles. Consequently a low-speed print takes twice as long as a high-speed print, since the high-speed print fires twice as many nozzles at once.

Figure 43 illustrates the composition of a phasegroup. The distance between adjacent podgroups is exaggerated for clarity.

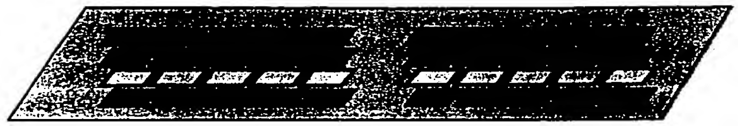


Figure 43. A single phasegroup contains 2 podgroups

14.1.1.5 2 Phasegroups Make a Firegroup

Two phasegroups (PhasegroupA and PhasegroupB) are organized into a single ***firegroup***, with 4 firegroups in each segment. Firegroups are so named because they all fire the same nozzles simultaneously. Two enable lines, AEnable and BEnable, allow the firing of PhasegroupA nozzles and PhasegroupB nozzles independently as different firing phases. The arrangement is shown in Figure 44. The distance between adjacent groupings is exaggerated for clarity.

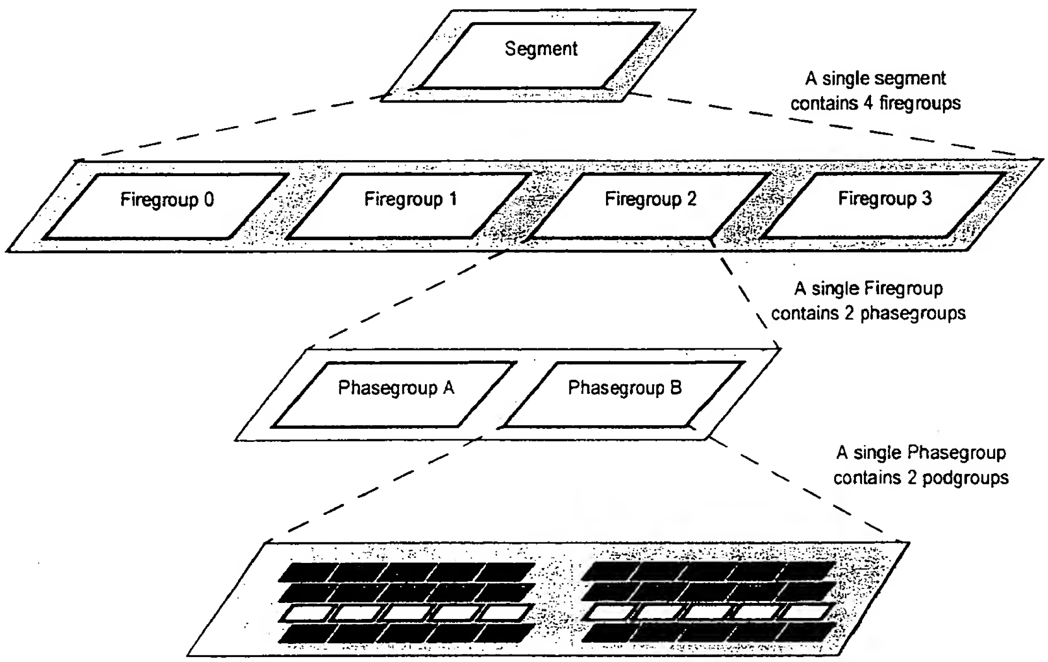


Figure 44. Relationship between Segments, Firegroups, Phasegroups, Podgroups and ChromaPods

14.1.1.6 *Nozzle Grouping Summary*

Table 10 is a summary of the nozzle groupings in a segment assuming a CMYK chromapod.

Table 10. Nozzle Groupings for a single segment

Nozzle	Base unit	1:1	1
Pod	Nozzles per pod	10:1	10
Chromapod	Pods per chromapod	C:1	10C
Podgroup	Chromapods per podgroup	5:1	50C
Phasegroup	Podgroups per phasegroup	2:1	100C
Firegroup	Phasegroups per firegroup	2:1	200C
Segment	Firegroups per segment	4:1	800C

The value of C, the number of colors contained in the segment, determines the total number of nozzles.

- With a 4 color segment, such as CMYK, the number of nozzles per segment is 3,200.
- With a 3 color segment, such as CMY, the number of nozzles per segment is 2,400.
- In a monochrome environment, the number of nozzles per segment is 800.

14.1.2 Load and Print Cycles

A single segment contains a total of 800C nozzles, where C is the number of colors in the segment. A *Print Cycle* involves the firing of up to all of these nozzles, dependent on the information to be printed. A *Load Cycle* involves the loading up of the segment with the information to be printed during the subsequent Print Cycle.

Each nozzle has an associated ***NozzleEnable*** bit that determines whether or not the nozzle will fire during the Print Cycle. The NozzleEnable bits (one per nozzle) are loaded via a set of shift registers.

Logically there are C shift registers per segment (one per color), each 800 deep. As bits are shifted into the shift register for a given color they are directed to the lower and upper nozzles on alternate pulses. Internally, each 800-deep shift register is comprised of two 400-deep shift registers: one for the upper nozzles, and one for the lower nozzles. Alternate bits are shifted into the alternate internal registers. As far as the external interface is concerned however, there is a single 800 deep shift register.

Once all the shift registers have been fully loaded (800 load pulses), all of the bits are transferred in parallel to the appropriate NozzleEnable bits. This equates to a single parallel transfer of 800C bits. Once the transfer has taken place, the Print Cycle can begin. The Print Cycle and the Load Cycle can occur simultaneously as long as the parallel load of all NozzleEnable bits occurs at the end of the Print Cycle.

14.1.2.1 *Load Cycle*

The Load Cycle is concerned with loading the segment's shift registers with the next Print Cycle's NozzleEnable bits.

Each segment has C inputs directly related to the C shift registers (where C is the number of colors in the segment). These inputs are named ***ColorNData***, where N is 1 to C (for example, a 4 color segment would have 4 inputs labeled ***Color1Data***, ***Color2Data***, ***Color3Data*** and ***Color4Data***). A single pulse on the ***SRClock*** line transfers C bits into the appropriate shift registers. Alternate pulses transfer bits to the lower and upper nozzles respectively. A total of 800 pulses are required for the complete transfer of data. Once all 800C bits have been transferred, a single pulse on the ***PTransfer*** line causes the parallel transfer of data from the shift registers to the appropriate NozzleEnable bits.

The parallel transfer via a pulse on PTransfer must take place *after* the Print Cycle has finished. Otherwise the NozzleEnable bits for the line being printed will be incorrect.

It is important to note that the odd and even dot outputs, although printed during the same Print Cycle, do not appear on the same physical output line. The physical separation of odd and even nozzles within the printhead, as well as separation between nozzles of different colors ensures that they will produce dots on different lines of the page. This relative difference must be accounted for when loading the data into the printhead. The actual difference in lines depends on the characteristics of the inkjet mechanism used in the printhead. The differences can be defined by variables D_1 and D_2 where D_1 is the distance between nozzles of different colors, and D_2 is the distance between nozzles of the same color. Table 11 shows the dots transferred to a C color segment on the first 4 pulses.

Table 11. Order of Dots Transferred to a Segment

Pulse	Dot	Color1 Line	Color2 Line	Color3 Line	ColorC Line
1	0	N	$N+D_1$[a]	$N+2D_1$	$N+(C-1)D_1$
2	1	$N+D_2$[b]	$N+D_1+D_2$	$N+2D_1+D_2$	$N+(C-1)D_1+D_2$
3	2	N	$N+D_1$	$N+2D_1$	$N+(C-1)D_1$
4	3	$N+D_2$	$N+D_1+D_2$	$N+2D_1+D_2$	$N+(C-1)D_1+D_2$

a. D_1 = number of lines between the nozzles of one color and the next (likely = 4-8)

b. D_2 = number of lines between two rows of nozzles of the same color (likely = 1)

And so on for all 800 pulses.

Data can be clocked into a segment at a maximum rate of 20 MHz, which will load the entire 800C bits of data in 40µs.

14.1.2.2 Print Cycle

A single Memjet printhead segment contains 800 nozzles. To fire them all at once would consume too much power and be problematic in terms of ink refill and nozzle interference. This problem is made more apparent when we consider that a Memjet printhead is composed of multiple 1/2 inch segments, each with 800 nozzles. Consequently two firing modes are defined: a low-speed printing mode and a high-speed printing mode:

- In the low-speed print mode, there are 200 phases, with each phase firing 4C nozzles (C per firegroup, where C is the number of colors).
- In the high-speed print mode, there are 100 phases, with each phase firing 8C nozzles, (2C per firegroup, where C is the number of colors).

The nozzles to be fired in a given firing pulse are determined by

- 3 bits ***ChromapodSelect*** (select 1 of 5 chromapods from a firegroup)
- 4 bits ***NozzleSelect*** (select 1 of 10 nozzles from a pod)
- 2 bits of ***PodgroupEnable*** lines (select 0, 1, or 2 podgroups to fire)

When one of the PodgroupEnable lines is set, only the specified Podgroup's 4 nozzles will fire as determined by ChromapodSelect and NozzleSelect. When both of the PodgroupEnable lines are set, both of the podgroups will fire their nozzles. For the low-speed mode, two fire pulses are required, with PodgroupEnable = 10 and 01 respectively. For the high-speed mode, only one fire pulse is required, with PodgroupEnable = 11.

The duration of the firing pulse is given by the ***AEnable*** and ***BEnable*** lines, which fire the PhasegroupA and PhasegroupB nozzles from all firegroups respectively. The typical duration of a firing pulse is 1.3 - 1.8 µs. The duration of a pulse depends on the viscosity of the ink (dependent on temperature and ink characteristics) and the amount of power available to the printhead. See Section 14.1.3 on page 61 for details on feedback from the printhead in order to compensate for temperature change.

The AEnable and BEnable are separate lines in order that the firing pulses can overlap. Thus the 200 phases of a low-speed Print Cycle consist of 100 A phases and 100 B phases, effectively giving 100 sets of Phase A and Phase B. Likewise, the 100 phases of a high-speed print cycle consist of 50 A phases and 50 B phases, effectively giving 50 phases of phase A and phase B.

Figure 45 shows the AEnable and BEnable lines during a typical Print Cycle. In a high-speed print there are 50 2µs cycles, while in a low-speed print there are 100 2µs cycles.

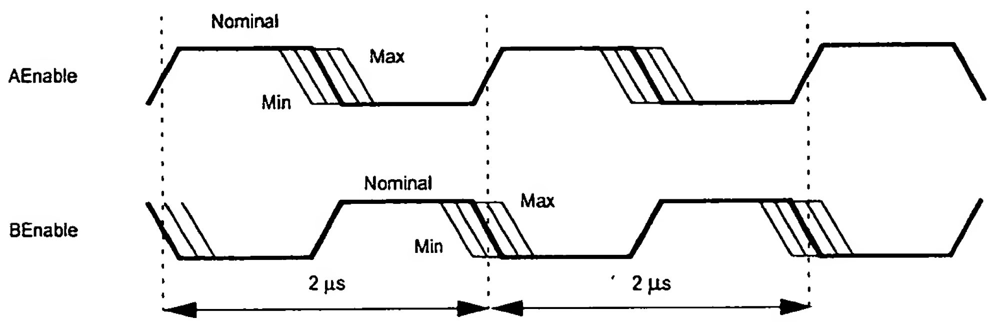


Figure 45. AEnable and BEnable During a Typical Print Cycle

For the high-speed printing mode, the firing order is:

- ChromapodSelect 0, NozzleSelect 0, PodgroupEnable 11 (Phases A and B)
- ChromapodSelect 1, NozzleSelect 0, PodgroupEnable 11 (Phases A and B)
- ChromapodSelect 2, NozzleSelect 0, PodgroupEnable 11 (Phases A and B)
- ChromapodSelect 3, NozzleSelect 0, PodgroupEnable 11 (Phases A and B)
- ChromapodSelect 4, NozzleSelect 0, PodgroupEnable 11 (Phases A and B)
- ChromapodSelect 0, NozzleSelect 1, PodgroupEnable 11 (Phases A and B)
- ...
- ChromapodSelect 3, NozzleSelect 9, PodgroupEnable 11 (Phases A and B)
- ChromapodSelect 4, NozzleSelect 9, PodgroupEnable 11 (Phases A and B)

For the low-speed printing mode, the firing order is similar. For each phase of the high speed mode where PodgroupEnable was 11, two phases of PodgroupEnable = 01 and 10 are substituted as follows:

- ChromapodSelect 0, NozzleSelect 0, PodgroupEnable 01 (Phases A and B)
- ChromapodSelect 0, NozzleSelect 0, PodgroupEnable 10 (Phases A and B)
- ChromapodSelect 1, NozzleSelect 0, PodgroupEnable 01 (Phases A and B)
- ChromapodSelect 1, NozzleSelect 0, PodgroupEnable 10 (Phases A and B)
- ...
- ChromapodSelect 3, NozzleSelect 9, PodgroupEnable 01 (Phases A and B)
- ChromapodSelect 3, NozzleSelect 9, PodgroupEnable 10 (Phases A and B)
- ChromapodSelect 4, NozzleSelect 9, PodgroupEnable 01 (Phases A and B)
- ChromapodSelect 4, NozzleSelect 9, PodgroupEnable 10 (Phases A and B)

When a nozzle fires, it takes approximately 100µs to refill. The nozzle cannot be fired before this refill time has elapsed. This limits the fastest printing speed to 100µs per line. In the high-speed print mode, the time to print a line is 100µs, so the time between firing a nozzle from one line to the next matches the refill time. The low-speed print mode is slower than this. so is also acceptable.

The firing of a nozzle also causes acoustic perturbations for a limited time within the common ink reservoir of that nozzle's pod. The perturbations can interfere with the firing of another nozzle within the same pod. Consequently, the firing of nozzles within a pod should be offset from each other as long as possible. We therefore fire four nozzles from a chromapod (one nozzle per color) and then move onto the next chromapod within the podgroup.

- In the low-speed printing mode the podgroups are fired separately. Thus the 5 chromapods within both podgroups must all fire before the first chromapod fires again, totalling $10 \times 2\mu s$ cycles. Consequently each pod is fired once per 20µs.
- In the high-speed printing mode, the podgroups are fired together. Thus the 5 chromapods within a single podgroups must all fire before the first chromapod fires again, totalling $5 \times 2\mu s$ cycles. Consequently each pod is fired once per 10 µs.

As the ink channel is 300µm long and the velocity of sound in the ink is around 1500m/s, the resonant frequency of the ink channel is 2.5MHz. Thus the low-speed mode allows 50 resonant cycles for the acoustic pulse to dampen, and the high-speed mode allows 25 resonant cycles. Consequently any acoustic interference is minimal in both cases.

14.1.3 Feedback from a Segment

A segment produces several lines of feedback. The feedback lines are used to adjust the timing of the firing pulses. Since multiple segments are collected together into a printhead, it is effective to share the feedback lines as a tri-state bus, with only one of the segments placing the feedback information on the feedback lines.

A pulse on the segment's ***SenseSegSelect*** line ANDed with data on ***Color1Data*** selects if the particular segment will provide the feedback. The feedback sense lines will come from that segment until the next SenseSegSelect pulse. The feedback sense lines are as follows:

- ***Tsense*** informs the controller how hot the printhead is. This allows the controller to adjust timing of firing pulses, since temperature affects the viscosity of the ink.
- ***Vsense*** informs the controller how much voltage is available to the actuator. This allows the controller to compensate for a flat battery or high voltage source by adjusting the pulse width.
- ***Rsense*** informs the controller of the resistivity (Ohms per square) of the actuator heater. This allows the controller to adjust the pulse widths to maintain a constant energy irrespective of the heater resistivity.
- ***Wsense*** informs the controller of the width of the critical part of the heater, which may vary up to ± 5% due to lithographic and etching variations. This allows the controller to adjust the pulse width appropriately.

14.1.4 Preheat Cycle

The printing process has a strong tendency to stay at the equilibrium temperature. To ensure that the first section of the printed photograph has a consistent dot size, the equilibrium temperature must be met *before* printing any dots. This is accomplished via a preheat cycle.

The Preheat cycle involves a single Load Cycle to all nozzles of a segment with 1s (i.e. setting all nozzles to fire), and a number of short firing pulses to each nozzle. The duration of the pulse must be insufficient to fire the drops, but enough to heat up the ink. Altogether about 200 pulses for each nozzle are required, cycling through in the same sequence as a standard Print Cycle.

Feedback during the Preheat mode is provided by Tsense, and continues until equilibrium temperature is reached (about 30° C above ambient). The duration of the Preheat mode is around 50 milliseconds, and depends on the ink composition.

Preheat is performed before each print job. This does not affect performance as it is done while the data is being transferred to the printer.

14.1.5 Cleaning Cycle

In order to reduce the chances of nozzles becoming clogged, a cleaning cycle can be undertaken before each print job. Each nozzle is fired a number of times into an absorbent sponge.

The cleaning cycle involves a single Load Cycle to all nozzles of a segment with 1s (i.e. setting all nozzles to fire), and a number of firing pulses to each nozzle. The nozzles are cleaned via the same nozzle firing sequence as a standard Print Cycle. The number of times that each nozzle is fired depends upon the ink composition and the time that the

printer has ben idle. As with preheat, the cleaning cycle has no effect on printer performance.

14.1.6 Printhead Interface Summary

Each segment has the following connections to the bond pads:

Table 12. Segment Interface Connections

Chromapod Select	3	Select which chromapod will fire (0-4)
NozzleSelect	4	Select which nozzle from the pod will fire (0-9)
PodgroupEnable	2	Enable the podgroups to fire (choice of: 01, 10, 11)
AEnable	1	Firing pulse for podgroup A
BEnable	1	Firing pulse for podgroup B
ColorNData	C	Input to shift registers (1 bit for each of C colors in the segment)
SRClock	1	A pulse on SRClock (ShiftRegisterClock) loads C bits from ColorData into the C shift registers.
PTransfer	1	Parallel transfer of data from the shift registers to the internal NozzleEnable bits (one per nozzle).
SenseSegSelect	1	A pulse on SenseSegSelect ANDed with data on Color1Data selects the sense lines for this segment.
Tsense	1	Temperature sense
Vsense	1	Voltage sense
Rsense	1	Resistivity sense
Wsense	1	Width sense
Logic GND	1	Logic ground
Logic PWR	1	Logic power
V-	21	Actuator Ground
V+	21	Actuator Power
TOTAL	62+C	(if C is 4, Total = 66)

14.2 MAKING MEMJET PRINTHEADS OUT OF SEGMENTS

A Memjet printhead is composed of a number of identical 1/2 inch printhead segments. These 1/2 inch segments are manufactured together or placed together after manufacture to produce a printhead of the desired length. Each 1/2 inch segments prints 800 1600 dpi bi-level dots in up to 4 colors over a different part of the page to produce the final image. Although each segment produces 800 dots of the final image, each dot is represented by a combination of colored inks.

A 4-inch printhead, for example, consists of 8 segments, typically manufactured as a monolithic printhead. In a typical 4-color printing application (cyan, magenta, yellow, black), each of the segments prints bi-level cyan, magenta, yellow and black dots over a different part of the page to produce the final image. The positions of the segments are shown in Figure 46.

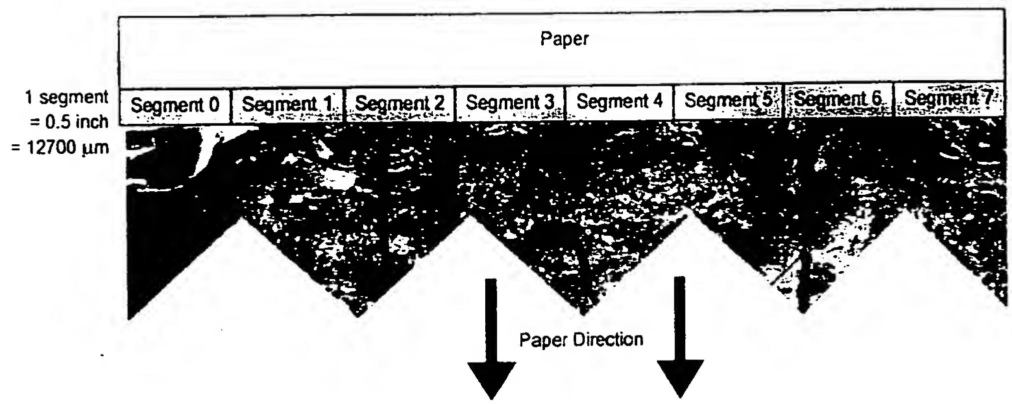


Figure 46. Arrangement of Segments in a 4-inch Printhead

An 8-inch printhead can be constructed from two 4-inch printheads or from a single 8-inch printhead consisting of 16 segments. Regardless of the construction mechanism, the effective printhead is still 8 inches in length.

A 2-inch printhead has a similar arrangement, but only uses 4 segments. Likewise, a full-bleed A4/Letter printer uses 17 segments for an effective 8½" printing area.

Since the total number of nozzles in a segment is 800C (see Table 10), the total number of nozzles in a given printhead with S segments is 800CS. Thus segment N is responsible for printing dots 800N to 800N+799.

A number of considerations must be made when wiring up a printhead. As the width of the printhead increases, the number of segments increases, and the number of connections also increases. Each segment has its own ColorData connections (C of them), as well as SRClock and other connections for loading and printing.

14.2.1 Loading Considerations

When the number of segments *S* is small it is reasonable to load all the segments simultaneously by using a common SRClock line and placing C bits of data on each of the ColorData inputs for the segments. In a 4-inch printer, S=8, and therefore the total number of bits to transfer to the printhead in a single SRClock pulse is 32. However for an 8-inch printer, S=16, and it is unlikely to be reasonable to have 64 data lines running from the print data generator to the printhead.

Instead, it is convenient to group a number of segments together for loading purposes. Each group of segments is small enough to be loaded simultaneously, and share an SRClock. For example, an 8-inch printhead can have 2 segment groups, each segment group containing 8 segments. 32 ColorData lines can be shared for both groups, with 2 SRClock lines, one per segment group.

When the number of segment groups is not easily divisible, it is still convenient to group the segments. One example is a 8½" printer for producing A4/Letter pages. There are 17 segments, and these can be grouped as two groups of 9 (9C bits of data going to each segment, with all 9C bits used in the first group, and only 8C bits used for the second group), or as 3 groups of 6 (again, C bits are unused in the last group).

As the number of segment groups increases, the time taken to load the printhead increases. When there is only one group, 800 load pulses are required (each pulse transfers C data bits). When there are G groups, 800G load pulses are required. The bandwidth of the connection between the data generator and the printhead must be able to cope and be within the allowable timing parameters for the particular application.

If ***G*** is the number of segment groups, and ***L*** is the largest number of segments in a group, the printhead requires LC ColorData lines and G SRClock lines. Regardless of G, only a single PTransfer line is required - it can be shared across all segments.

Since L segments in each segment group are loaded with a single SRClock pulse, any printing process must produce the data in the correct sequence for the printhead. As an example, when G=2 and L=4, the first SRClock1 pulse will transfer the ColorData bits for the next Print Cycle's dot 0, 800, 1600, and 2400. The first SRClock2 pulse will transfer the ColorData bits for the next Print Cycle's dot 3200, 4000, 4800, and 5600. The second SRClock1 pulse will transfer the ColorData bits for the next Print Cycle's dot 1, 801, 1601, and 2401. The second SRClock2 pulse will transfer the ColorData bits for the next Print Cycle's dot 3201, 4001, 4801 and 5601.

After 800G SRClock pulses (800 to each of SRClock1 and SRClock2), the entire line has been loaded into the printhead, and the common PTransfer pulse can be given.

It is important to note that the odd and even color outputs, although printed during the same Print Cycle, do not appear on the same physical output line. The physical separation of odd and even nozzles within the printhead, as well as separation between nozzles of different colors ensures that they will produce dots on different lines of the page. This relative difference must be accounted for when loading the data into the printhead. The actual difference in lines depends on the characteristics of the inkjet mechanism used in the printhead. The differences can be defined by variables D_1 and D_2 where D_1 is the distance between nozzles of different colors, and D_2 is the distance between nozzles of the same

color. Considering only a single segment group, Table 13 shows the dots transferred to segment *n* of a printhead during the first 4 pulses of the shared SRClock.

Table 13. Order of Dots Transferred to a Segment in a Printhead

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
1	800S[a]	N	$N+D_1$[b]	$N+(C-1)D_1$
2	800S+1	$N+D_2$[c]	$N+D_1+D_2$	$N+(C-1)D_1+D_2$
3	800S+2	N	$N+D_1$	$N+(C-1)D_1$
4	800S+3	$N+D_2$	$N+D_1+D_2$	$N+(C-1)D_1+D_2$

a. S = segment number

b. D_1 = number of lines between the nozzles of one color and the next (likely = 4-8)

c. D_2 = number of lines between two rows of nozzles of the same color (likely = 1)

And so on for all 800 SRClock pulses to the particular segment group.

14.2.2 Printing Considerations

With regards to printing, we print 4C nozzles from each segment in the low-speed printing mode, and 8C nozzles from each segment in the high speed printing mode.

While it is certainly possible to wire up segments in any way, this document only considers the situation where all segments fire simultaneously. This is because the low-speed printing mode allows low-power printing for small printheads (e.g. 2-inch and 4-inch), and the controller chip design assumes there is sufficient power available for the large print sizes (such as 8-18 inches). It is a simple matter to alter the connections in the printhead to allow grouping of firing should a particular application require it.

When all segments are fired at the same time 4CS nozzles are fired in the low-speed printing mode and 8CS nozzles are fired in the high-speed printing mode. Since all segments print simultaneously, the printing logic is the same as defined in Section 14.1.2.2 on page 58.

The timing for the two printing modes is therefore:

- 200 µs to print a line at low speed (comprised of 100 2µs cycles)
- 100 µs to print a line at high speed (comprised of 50 2µs cycles)

14.2.3 Feedback Considerations

A segment produces several lines of feedback, as defined in Section 14.1.3 on page 61. The feedback lines are used to adjust the timing of the firing pulses. Since multiple segments are collected together into a printhead, it is effective to share the feedback lines as a tri-state bus, with only one of the segments placing the feedback information on the feedback lines at a time.

Since the selection of which segment will place the feedback information on the shared Tsense, Vsense, Rsense, and Wsense lines uses the Color1Data line, the groupings of segments for loading data can be used for selecting the segment for feedback.

Just as there are G SRClock lines (a single line is shared between segments of the same segment group), there are G SenseSegSelect lines shared in the same way. When the correct SenseSegSelect line is pulsed, the segment of that group whose Color1Data bit is set will start to place data on the shared feedback lines. The segment previously active in terms of feedback must also be disabled by having a 0 on its Color1Data bit, and this segment may be in a different segment group. Therefore when there is more than one segment group. changing the feedback segment requires two steps: disabling the old segment, and enabling the new segment.

14.2.4 Printhead Connection Summary

This section assumes that a printhead has been constructed from a number of segments as described in the previous sections. It assumes that for data loading purposes, the segments have been grouped into ***G*** segment groups, with ***L*** segments in the largest segment group. It assumes there are ***C*** colors in the printhead. It assumes that the firing mechanism for the printhead is that all segments fire simultaneously, and only one segment at a time places feedback information on a common tri-state bus. Assuming all these things, Table 14 lists the external connections that are available from a printhead:

Table 14. Printhead Connections

Name	#Pins	Description
ChromapodSelect	3	Select which chromapod will fire (0-4)
NozzleSelect	4	Select which nozzle from the pod will fire (0-9)
PodgroupEnable	2	Enable the podgroups to fire (choice of: 01, 10, 11)
AEnable	1	Firing pulse for phasegroup A
BEnable	1	Firing pulse for phasegroup B
ColorData	CL	Inputs to C shift registers of segments 0 to L-1
SRClock	G	A pulse on SRClock[N] (ShiftRegisterClock N) loads the current values from ColorData lines into the L segments in segment group N.
PTransfer	1	Parallel transfer of data from the shift registers to the internal NozzleEnable bits (one per nozzle).
SenseSegSelect	G	A pulse on SenseSegSelect N ANDed with data on Color1Data[n] selects the sense lines for segment n in segment group N.
Tsense	1	Temperature sense
Vsense	1	Voltage sense
Rsense	1	Resistivity sense
Wsense	1	Width sense
Logic GND	1	Logic ground
Logic PWR	1	Logic power
V-	Bus bars	Actuator Ground
V+		Actuator Power
TOTAL	18+2G+CL	

15 Memjet Printhead Interface

The printhead interface (PHI) is the means by which the processor loads the Memjet printhead with the dots to be printed, and controls the actual dot printing process. The PHI contains:

- a LineSyncGen unit (LSGU), which provides synchronization signals for multiple chips (allows side-by-side printing and front/back printing) as well as stepper motors.
- a Memjet interface (MJI), which transfers data to the Memjet printhead, and controls the nozzle firing sequences during a print.
- a line loader/format unit (LLFU) which loads the dots for a given print line into local buffer storage and formats them into the order required for the Memjet printhead.

The units within the PHI are controlled by a number of registers that are programmed by the processor. In addition, the processor is responsible for setting up the appropriate parameters in the DMA controller for the transfers from memory to the LLFU. This includes loading white (all 0's) into appropriate colors during the start and end of a page so that the page has clean edges.

The PHI is capable of dealing with a variety of printhead lengths and formats. In terms of broad operating customizations, the PHI is parameterized as follows:

Table 15. Basic Printing Parameters

Name	Description	Range
MaxColors	No of Colors in printhead	1-4
SegmentsPerXfer	No of segments written to per transfer. Is equal to the number of segments in the largest segment group	1-9
SegmentGroups	No of segment groups in printhead	1-4

The internal structure of the PHI allows for a maximum of 4 colors, 9 segments per transfer, and 4 transfers. Transferring 4 colors to 9 segments is 36 bits per transfer, and 4 transfers to 9 segments equates to a maximum printed line length of 18 inches. The total number of dots per line printed by an 18-inch 4 color printhead is 115,200 ($18 \times 1600 \times 4$).

Other example settings are shown in Table 16:

Table 16. Example Settings for Basic Printing Parameters

Printer Length	Printer Type	MaxColors	SegmentsPerXfer	SegmentGroups	BitsPerXfer	Comments
4 inch CMY	Photo	3	8	1	24	
8 inch CMYK	A4/Letter	4	8	2	32	
8½ inch CMYK	A4/Letter full bleed	4	9	2	36	Last xfer not fully used
12 inch CMYK	A4 long / A3 short	4	8	3	32	
16 inch CMYK		4	8	4	32	
17 inch CMYK	A3 long full bleed	4	9	4	36	Last xfer not fully used
18 inch CMYK		4	9	4	36	

15.1 BLOCK DIAGRAM OF PRINTHEAD INTERFACE

The internal structure of the Printhead Interface is shown in Figure 47.

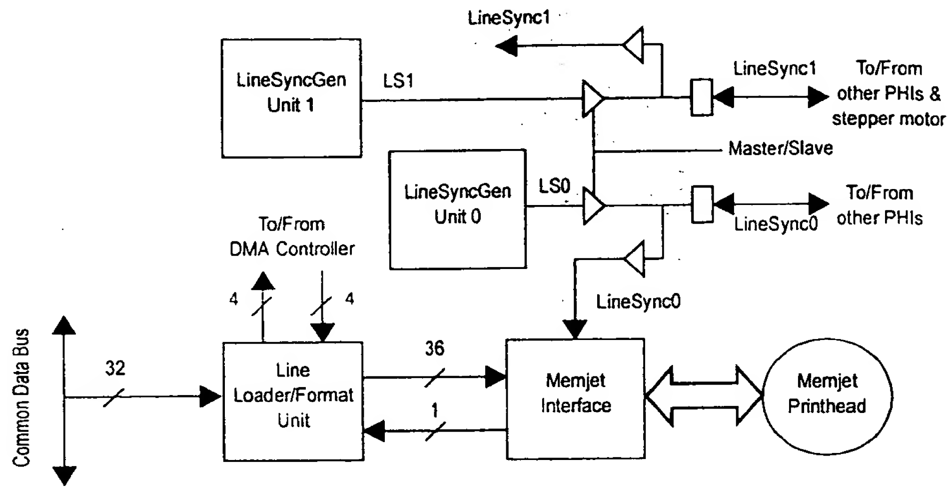


Figure 47. Internal Structure of Printhead Interface

In the PHI there are two LSGUs. The first LSGU produces LineSync0, which is used to control the Memjet Interface in all synchronized chips. The second LSGU produces LineSync1 which is used to pulse the paper drive stepper motor.

The Master/Slave pin on the chip allows multiple chips to be connected together for side-by-side printing, front/back printing etc. via a Master/Slave relationship. When the Master/Slave pin is attached to V_{DD}, the chip is considered to be the Master, and LineSync pulses generated by the two LineSyncGen units are enabled onto the two tri-state LineSync common lines (LineSync0 and LineSync1, shared by all the chips). When the Master/Slave pin is attached to GND, the chip is considered to be the Slave, and LineSync pulses generated by the two LineSyncGen units are not enabled onto the common LineSync lines. In this way, the Master chip's LineSync pulses are used by all PHIs on all the connected chips.

The following sections detail the LineSyncGen Unit, the Line Loader/Format Unit and Memjet Interface respectively.

15.2 LINESYNCGEN UNIT

The LineSyncGen units (LSGU) are responsible for generating the synchronization pulses required for printing a page. Each LSGU produces an external LineSync signal to enable line synchronization. The generator inside the LGSU generates a LineSync pulse when told to 'go', and then every so many cycles until told to stop. The LineSync pulse defines the start of the next line.

The exact number of cycles between LineSync pulses is determined by the CyclesBetweenPulses register, one per generator. It must be at least long enough to allow one line to print (100 μs or 200 μs depending on whether the speed is low or high) and another line to load, but can be longer as desired (for example, to accommodate special requirements of paper transport circuitry). If the CyclesBetweenPulses register is set to a number less than a line

print time, the page will not print properly since each LineSync pulse will arrive before the particular line has finished printing.

The following interface registers are contained in the LSGU:

Table 17. LineSyncGen Unit Registers

CyclesBetweenPulses	The number of cycles to wait between generating one LineSync pulse and the next.
Go	Controls whether the LSGU is currently generating LineSync pulses or not. A write of 1 to this register generates a LineSync pulse, transfers CyclesBetweenPulses to CyclesRemaining, and starts the countdown. When CyclesRemaining hits 0, another LineSync pulse is generated, CyclesBetweenPulses is transferred to CyclesRemaining and the countdown is started again. A write of 0 to this register stops the countdown and no more LineSync pulses are generated.
CyclesRemaining	A status register containing the number of cycles remaining until the next LineSync pulse is generated.

The LineSync pulse is not used directly from the LGSU. The LineSync pulse is enabled onto a tri-state LineSync line only if the Master/Slave pin is set to Master. Consequently the LineSync pulse is only used in the form as generated by the Master chip (pulses generated by Slave chips are ignored).

15.3 MEMJET INTERFACE

The Memjet interface (MJI) transfers data to the Memjet printhead, and controls the nozzle firing sequences during a print.

The MJI is simply a State Machine (see Figure 48) which follows the printhead loading and firing order described in Section 14.2.1 on page 64, Section 14.2.2 on page 65, and includes the functionality of the Preheat Cycle and Cleaning Cycle as described in Section 14.1.4 on page 61 and Section 14.1.5 on page 61. Both high-speed and low-speed printing modes are available, although the MJI always fires a given nozzle from all segments in a printhead simultaneously (there is no separate firing of nozzles from one segment and then others). Dot counts for each color are also kept by the MJI.

The MJI loads data into the printhead from a choice of 2 data sources:

- All 1s. This means that all nozzles will fire during a subsequent Print cycle, and is the standard mechanism for loading the printhead for a preheat or cleaning cycle.
- From the 36-bit input held in the Transfer register of the LLFU. This is the standard means of printing an image. The 36-bit value from the LLFU is directly sent to the printhead and a 1-bit 'Advance' control pulse is sent to the LLFU.

The MJI knows how many lines it has to print for the page. When the MJI is told to 'go', it waits for a LineSync pulse before it starts the first line. Once it has finished loading/printing a line, it waits until the next LineSync pulse before starting the next line. The MJI stops once the specified number of lines has been loaded/printed, and ignores any further LineSync pulses.

The MJI is therefore directly connected to the LLFU, LineSync0 (shared between all synchronized chips), and the external Memjet printhead. The basic structure is shown in Figure 48.

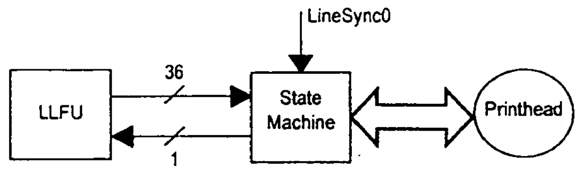


Figure 48. Memjet Interface

The MJI accepts 36 bits of data from the LLFU. Of these 36 bits, only the bits corresponding to the number of segments and number of colors will be valid. For example, if there are only 2 colors and 9 segments, bits 0-1 will be valid for segment 0, bits 2-3 will be invalid, bits 4-5 will be valid for segment 1, bits 6-7 will be invalid etc. The state machine does not care which bits are valid and which bits are not valid - it merely passes the bits out to the printhead. The data lines and control signals coming out of the MJI can be wired appropriately to the pinouts of the chip, using as few pins as required by the application range of the chip (see Section 15.3.1 on page 70 for more information).

15.3.1 Connections to Printhead

The MJI has a number of connections to the printhead, including a maximum of 4 colors, clocked in to a maximum of 9 segments per transfer to a maximum of 4 segment groups. The lines coming from the MJI can be directly connected to pins on the chip, although not all lines will always be pins. For example, if the chip is specifically designed for only connecting to 8 inch CMYK printers, only 32 bits of data need to be transferred each transfer pulse. Consequently 32 pins of data out (8 pins per color), and not 36 pins are required. In the same way, only 2 SRClock pulses are required, so only 2 pins instead of 4 pins are required to cater for the different SRClocks. And so on.

If the chip must be completely generic, then all connections from the MJI must be connected to pins on the chip (and thence to the Memjet printhead).

Table 18 lists the maximum connections from the MJI, many of which are always connected to pins on the chip. Where the number of pins is variable, a footnote explains what the number of pins depends upon. The sense of input and output is with respect to the MJI. The names correspond to the pin connections on the printhead.

Table 18. Memjet Interface Connections

Name	#Pins	I/O	Description
Chromapod Select	3	O	Select which chromapod will fire (0-4)
NozzleSelect	4	O	Select which nozzle from the pod will fire (0-9)
PodgroupEnable	2	O	Enable the podgroups to fire (choice of: 01, 10, 11)
AEnable	1	O	Firing pulse for podgroup A. In the current design all segments fire simultaneously, although multiple AEnable lines could be added for dividing the firing sequence over multiple segment groups for reasons of power and speed.

Table 18. Memjet Interface Connections

BEnable	1	O	Firing pulse for podgroup B. In the current design all segments fire simultaneously, although multiple BEnable lines could be added for dividing the firing sequence over multiple segment groups for reasons of power and speed.
Color1Data[0-8]	9[a]	O	Output to Color1Data shift register of segments 0-8
Color2Data[0-8]	9[b]	O	Output to Color2Data shift register of segments 0-8
Color3Data[0-8]	9[c]	O	Output to Color3Data shift register of segments 0-8
Color4Data[0-8]	9[d]	O	Output to Color4Data shift register of segments 0-8
SRClock[1-4]	4[e]	O	A pulse on SRClock[N] (ShiftRegisterClock) loads the current values from Color1Data[0-8], Color2Data[0-8], Color3Data[0-8] and Color4Data[0-8] into the segment group N on the printhead.
PTransfer	1	O	Parallel transfer of data from the shift registers to the printhead's internal NozzleEnable bits (one per nozzle).
SenseSegSelect[1-4]	4[f]	O	A pulse on SenseSegSelect[N] ANDed with data on Color1Data[n] enables the sense lines for segment n in segment group N of the printhead.
Tsense	1	I	Temperature sense
Vsense	1	I	Voltage sense
Rsense	1	I	Resistivity sense
Wsense	1	I	Width sense
TOTAL	52		

a. Although 9 lines are available from the MJI, the number of pins coming from the chip will only reflect the actual number of segments in a segment group. The pins for Color1Data are mandatory, since each printhead must print in at *least* 1 color.

b. These lines are only translated into pins if the chip is to control a printhead with at least 2 colors. Although 9 lines are available from the MJI, the number of pins coming from the chip for Color2Data will only reflect the *actual* number of segments in a segment group.

c. These lines are only translated into pins if the chip is to control a printhead with at least 3 colors. Although 9 lines are available from the MJI, the number of pins coming from the chip for Color3Data will only reflect the *actual* number of segments in a segment group.

d. These lines are only translated into pins if the chip is to control a printhead with 4 colors. Although 9 lines are available from the MJI, the number of pins coming from the chip for Color4Data will only reflect the *actual* number of segments in a segment group.

e. Although 4 lines are available from the MJI, the number of pins coming from the chip will only reflect the *actual* number of segment groups. A minimum of 1 pin is required since there is at least 1 segment group (the entire printhead).

f. Although 4 lines are available from the MJI, the number of pins coming from the chip will only reflect the *actual* number of segment groups. A minimum of 1 pin is required since there is at least 1 segment group (the entire printhead).

15.3.2 Firing Pulse Duration

The duration of firing pulses on the AEnable and BEnable lines depend on the viscosity of the ink (which is dependant on temperature and ink characteristics) and the amount of power available to the printhead. The typical pulse duration range is 1.3 to 1.8 μs. The MJI therefore contains a programmable pulse duration table, indexed by feedback from the printhead. The table of pulse durations allows the use of a lower cost power supply, and aids in maintaining more accurate drop ejection.

The Pulse Duration table has 256 entries, and is indexed by the current Vsense and Tsense settings. The upper 4-bits of address come from Vsense, and the lower 4-bits of address

come from Tsense. Each entry is 8 bits, and represents a fixed point value in the range of 0-4μs. The process of generating the AEnable and BEnable lines is shown in Figure 49.

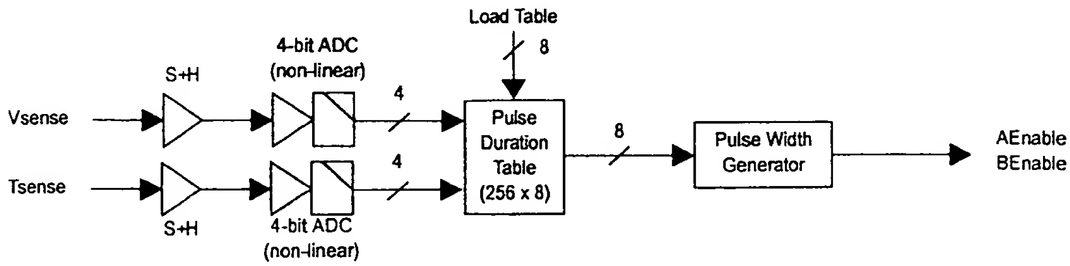


Figure 49. Generation of AEnable and BEnable Pulse Widths

The 256-byte table is written by the CPU before printing the first page. The table may be updated in between pages if desired. Each 8-bit pulse duration entry in the table combines:

- User brightness settings (from the page description)
- Viscosity curve of ink (from the QA Chip)
- Rsense
- Wsense
- Vsense
- Tsense

15.3.3 Dot Counts

The MJI maintains a count of the number of dots of each color fired from the printhead. The dot count for each color is a 32-bit value, individually cleared under processor control. At 32-bits length, each dot count can hold a maximum coverage dot count of 17 8-inch × 12-inch pages, although in typical usage, the dot count will be read and cleared after each page or half-page.

The dot counts are used by the processor to update the QA chip in order to predict when the ink cartridge runs out of ink. The processor knows the volume of ink in the cartridge for each of the colors from the QA chip. Counting the number of drops eliminates the need for ink sensors, and prevents the ink channels from running dry. An updated drop count is written to the QA chip after each page. A new page will not be printed unless there is enough ink left, and allows the user to change the ink without getting a dud half-printed page which must be reprinted.

The layout of the dot counter for Color1 is shown in Figure 50. The remaining 3 dot counters (Color1DotCount, Color2DotCount, and Color3DotCount) are identical in structure.

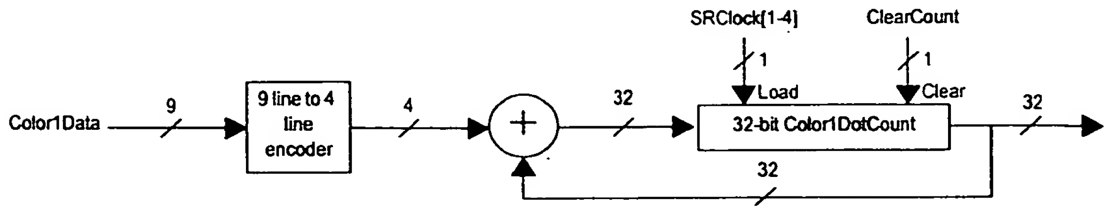


Figure 50. Dot Count Logic

15.3.4 Registers

The processor communicates with the MJI via a register set. The registers allow the processor to parameterize a print as well as receive feedback about print progress.

The following registers are contained in the MJI:

Table 19. Memjet Interface Registers

Register Name	Description
Print Parameters	
SegmentsPerXfer	The number of segments to write to each transfer. This also equals the number of cycles to wait between each transfer (before generating the next Advance pulse). Each transfer has MaxColors x SegmentsPerXfer valid bits.
SegmentGroups	The number of segment groups in the printhead. This equals the number of times that SegmentsPerXfer cycles must elapse before a single dot has been written to each segment of the printhead. The MJI does this 800 times to completely transfer all the data for the line to the printhead.
PrintSpeed	Whether to print at low or high speed (determines the value on the PodgroupEnable lines during the print).
NumLines	The number of Load/Print cycles to perform.
Monitoring the Print (read only from point of view of processor)	
Status	The Memjet Interface's Status Register
LinesRemaining	The number of lines remaining to be printed. Only valid while Go=1. Starting value is NumLines and counts down to 0.
TransfersRemaining	The number of sets of SegmentGroups transfers remaining before the Printhead is considered loaded for the current line. Starts at 800 and counts down to 0. Only valid while Go=1.
SegGroupsRemaining	The number of segment groups remaining in the current set of transfers of 1 dot to each segment. Starts at SegmentGroups and counts down to 0. Only valid while Go=1.
SenseSegment	The 9-bit value to place on the Color1Data lines during a subsequent feedback SenseSegSelect pulse. Only 1 of the 9 bits should be set, corresponding to one of the (maximum) 9 segments. See SenseSelect for how to determine which of the segment groups to sense.

Table 19. Memjet Interface Registers

SetAllNozzles	If non-zero, the 36-bit value written to the printhead during the LoadDots process is all 1s, so that all nozzles will be fired during the subsequent PrintDots process. This is used during the preheat and cleaning cycles. If 0, the 36-bit value written to the printhead comes from the LLFU. This is the case during the actual printing of regular images.
Actions	
Reset	A write to this register resets the MJI, stops any loading or printing processes, and loads all registers with 0.
SenseSelect	A write to this register with any value clears the FeedbackValid bit of the Status register, and the remaining action depends on the values in the LoadingDots and PrintingDots status bits. If either of the status bits are set, the Feedback bit is cleared and nothing more is done. If both status bits are clear, a pulse is given simultaneously on all 4 SenseSegSelect lines with all ColorData bits 0. This stops any existing feedback. Depending on the two low-order bits written to SenseSelect register, a pulse is given on SenseSegSelect1, SenseSegSelect2, SenseSegSelect3, or SenseSegSelect4 line, with the Color1Data bits set according to the SenseSegment register. Once the various sense lines have been tested, the values are placed in the Tsense, Vsense, Rsense, and Wsense registers, and the Feedback bit of the Status register is set.
Go	A write of 1 to this bit starts the LoadDots / PrintDots cycles, which commences with a wait for the first LineSync pulse. A total of NumLines lines are printed, each line being loaded/printed after the receipt of a LineSync pulse. The loading of each line consists of SegmentGroups 36-bit transfers. As each line is printed, LinesRemaining decrements, and TransfersRemaining is reloaded with SegmentGroups again. The status register contains print status information. Upon completion of NumLines, the loading/printing process stops, the Go bit is cleared, and any further LineSync pulses are ignored. During the final print cycle, nothing is loaded into the printhead. A write of 0 to this bit stops the print process, but does not clear any other registers.
ClearCounts	A write to this register clears the Color1DotCount, Color2DotCount, Color3DotCount, and Color4DotCount registers if bits 0, 1, 2, or 3 respectively are set. Consequently a write of 0 has no effect.
Feedback	
Tsense	Read only feedback of Tsense from the last SenseSegSelect pulse sent to segment SenseSegment. Is only valid if the FeedbackValid bit of the Status register is set.
Vsense	Read only feedback of Vsense from the last SenseSegSelect pulse sent to segment SenseSegment. Is only valid if the FeedbackValid bit of the Status register is set.
Rsense	Read only feedback of Rsense from the last SenseSegSelect pulse sent to segment SenseSegment. Is only valid if the FeedbackValid bit of the Status register is set.
Wsense	Read only feedback of Wsense from the last SenseSegSelect pulse sent to segment SenseSegment. Is only valid if the FeedbackValid bit of the Status register is set.
Color1DotCount	Read only 32-bit count of color1 dots sent to the printhead.
Color2DotCount	Read only 32-bit count of color2 dots sent to the printhead.

Table 19. Memjet Interface Registers

Register	[illegible]
Color3DotCount	Read only 32-bit count of color3 dots sent to the printhead
Color4DotCount	Read only 32-bit count of color4 dots sent to the printhead

The MJI's Status Register is a 16-bit register with bit interpretations as follows:

Table 20. MJI Status Register

Name	[illegible]	Description
LoadingDots	1	If set, the MJI is currently loading dots, with the number of dots remaining to be transferred in TransfersRemaining. If clear, the MJI is not currently loading dots
PrintingDots	1	If set, the MJI is currently printing dots. If clear, the MJI is not currently printing dots.
PrintingA	1	This bit is set while there is a pulse on the AEnable line
PrintingB	1	This bit is set while there is a pulse on the BEnable line
FeedbackValid	1	This bit is set while the feedback values Tsense, Vsense, Rsense, and Wsense are valid.
Reserved	3	-
PrintingChromapod	4	This holds the current chromapod being fired while the PrintingDots status bit is set.
PrintingNozzles	4	This holds the current nozzle being fired while the PrintingDots status bit is set.

The following pseudocode illustrates the logic required to load a printhead for a single line. Note that loading commences only after the LineSync pulse arrives. This is to ensure the data for the line has been prepared by the LLFU and is valid for the first transfer to the printhead.

```
Wait for LineSync
For TransfersRemaining = 800 to 0
   For I = 0 to SegmentGroups
      If (SetAllNozzles)
         Set all ColorData lines to be 1
      Else
         Place 36 bit input on 36 ColorData lines
      EndIf
      Pulse SRClock[I]
      Wait SegmentsPerXfer cycles
      Send ADVANCE signal
   EndFor
EndFor
```

15.3.5 Preheat and Cleaning Cycles

The Cleaning and Preheat cycles are simply accomplished by setting appropriate registers in the MJI:

- SetAllNozzles = 1
- Set the PulseDuration register to either a low duration (in the case of the preheat mode) or to an appropriate drop ejection duration for cleaning mode.
- Set NumLines to be the number of times the nozzles should be fired
- Set the Go bit and then wait for the Go bit to be cleared when the print cycles have completed.

The LSGU must also be programmed to send LineSync pulses at the correct frequency.

15.4 LINE LOADER/FORMAT UNIT

The line loader/format unit (LLFU) loads the dots for a given print line into local buffer storage and formats them into the order required for the Memjet printhead. It is responsible for supplying the pre-calculated nozzleEnable bits to the Memjet interface for the eventual printing of the page.

The printing uses a double buffering scheme for preparing and accessing the dot-bit information. While one line is being loaded into the first buffer, the pre-loaded line in the second buffer is being read in Memjet dot order. Once the entire line has been transferred from the second buffer to the printhead via the Memjet interface, the reading and writing processes swap buffers. The first buffer is now read and the second buffer is loaded up with the new line of data. This is repeated throughout the printing process, as can be seen in the conceptual overview of Figure 51.

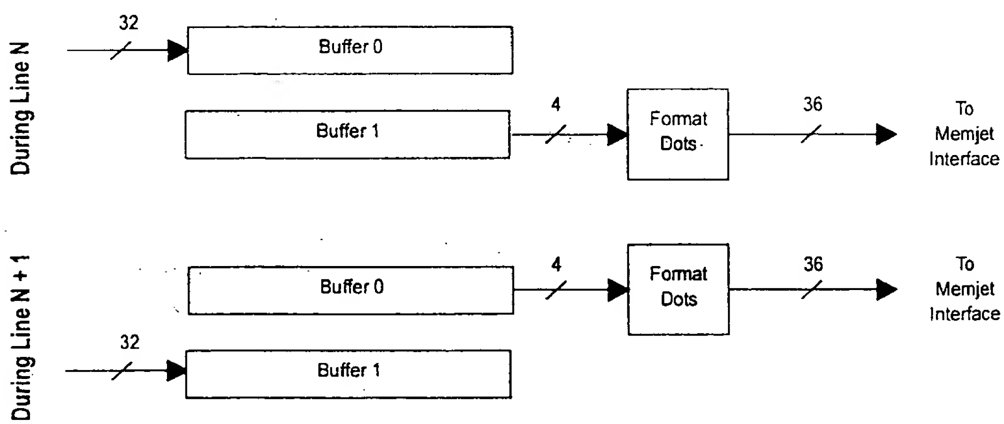


Figure 51. Conceptual Overview of Double Buffering During Print Lines N and N+1

The size of each buffer is 14KBytes to cater for the maximum line length of 18 inches in 4 colors ($18 \times 1600 \times 4$ bits = 115,200 bits = 14,400 bytes). The size for both Buffer 0 and Buffer 1 is 28.128 KBytes. While this design allows for a maximum print length of 18 inches, it is trivial to reduce the buffer size to target a specific application.

The actual implementation of the LLFU is shown in Figure 52. Since one buffer is being read from while the other is being written to, two sets of address lines must be used. The 32-bits DataIn from the common data bus are loaded depending on the WriteEnables, which are generated by the State Machine in response to the DMA Acknowledges.

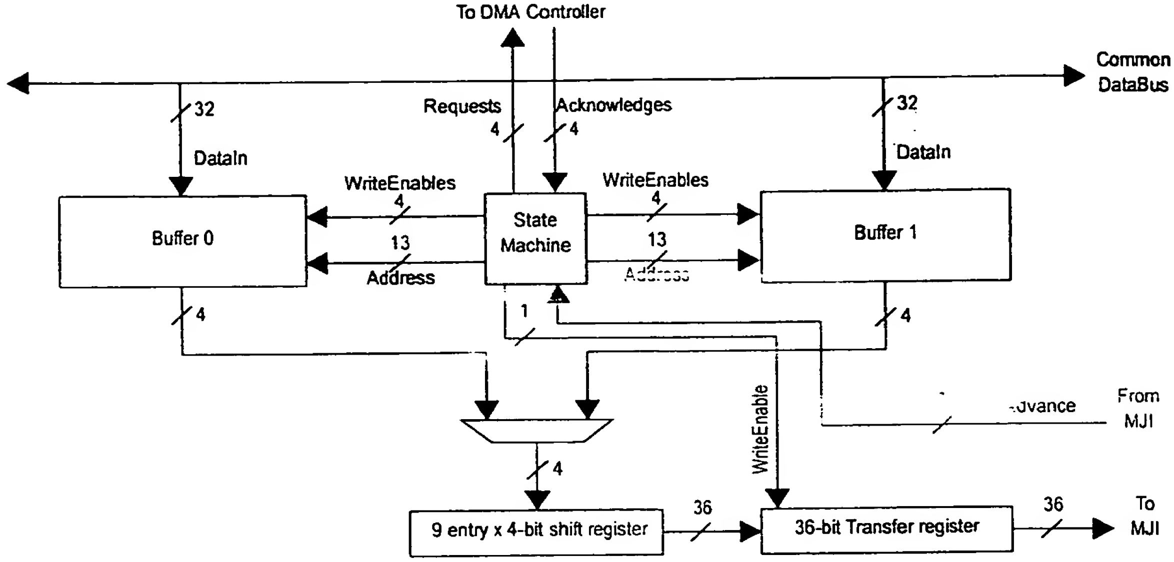


Figure 52. Structure of LLFU

A multiplexor chooses between the two 4-bit outputs of Buffer 0 and Buffer 1, and sends the result to a 9-entry by 4-bit shift register. After a maximum of 9 read cycles (the number depends on the number of segments written to per transfer), and whenever an Advance pulse comes from the MJI, the current 36-bit value from the shift register is gated into the 36-bit Transfer register, where it can be used by the MJI.

Note that not all the 36 bits are necessarily valid. The number of valid bits of 36 depends on the number of colors in the printhead, the number of segments, and the breakup of segment groups (if more than one segment group). For more information, see Section 14.2 on page 63.

A single line in an *L*-inch *C*-color printhead consists of ***1600L C***-color dots. At 1 bit per colored dot, a single print-line consists of ***1600LC bits***. The LLFU is capable of addressing a maximum line size of 18 inches in 4 colors, which equates to 108,800 bits (14 KBytes) per line. These bits must be supplied to the MJI in the correct order for being sent on to the printhead. See Section 14.2.1 on page 64 for more information concerning the Load Cycle dot loading order, but in summary, 2LC bits are transferred to the printhead in ***SegmentGroups*** transfers, with a maximum of 36 bits per transfer. Each transfer to a particular segment of the printhead must load all colors simultaneously.

15.4.1 Buffers

Each of the two buffers is broken into 4 sub-buffers, 1 per color. The size of each sub-buffer is 3600 bytes, enough to hold 18-inches of single color dots at 1600 dpi. The memory is accessed 32-bits at a time, so there are 900 addresses for each buffer (requiring 10 bits of address).

All the even dots are placed before the odd dots in each color's buffer, as shown in Figure 53. If there is any unused space it is placed at the end of each color's buffer.

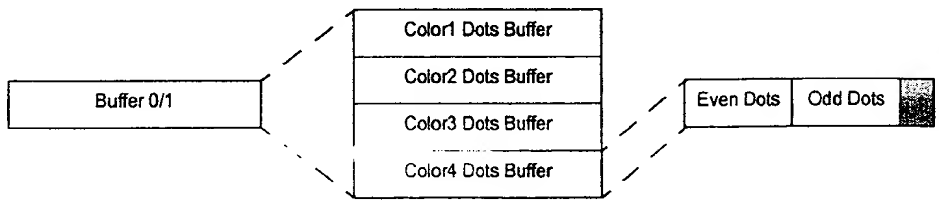


Figure 53. Conceptual Structure of Buffer

The *amount of memory* actually used is directly related to the printhead length. If the printhead is 18 inches, there are 1800 bytes of even dots followed by 1800 bytes of odd dots, with no unused space. If the printhead is 12 inches, there are 1200 bytes of even dots followed by 1200 odd dots, and 1200 bytes unused.

The *number of sub-buffers* gainfully used is directly related to the number of colors in the printhead. This number is typically 3 or 4, although it is quite feasible for this system to be used in a 1 or 2 color system (with some small memory wastage). In a desktop printing environment, the number of colors would be 4: Color1=Cyan, Color2=Magenta, Color3=Yellow, Color4=Black.

The addressing decoding circuitry is such that in a given cycle, a single 32-bit access can be made to all 4 sub-buffers - either a read from all 4 or a write to one of the 4. Only one bit of the 32-bits read from each color buffer is selected, for a total of 4 output bits. The process is shown in Figure 54. 15 bits of address allow the reading of a particular bit by means of 10-bits of address being used to select 32 bits, and 5-bits of address choose 1-bit from those 32. Since all color buffers share this logic, a single 15-bit address gives a total of 4 bits out, one per color. Each buffer has its own WriteEnable line, to allow a single

32-bit value to be written to a particular color buffer in a given cycle. The 32-bits of DataIn are shared, since only one buffer will actually clock the data in.

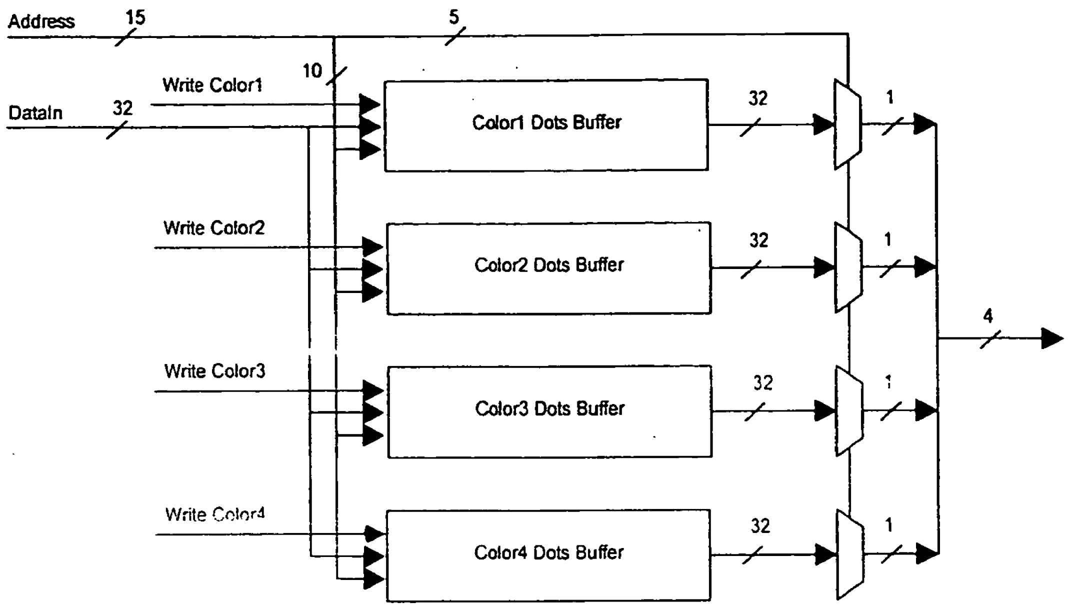


Figure 54. Logical Structure of Buffer

Note that regardless of the number of colors in the printhead, 4 bits are produced in a given read cycle (one bit from each color's buffer).

15.4.2 Address Generation

15.4.2.1 Reading

Address Generation for reading is straightforward. Each cycle we generate a bit address which is used to fetch 4 bits representing 1-bit per color for a particular segment. By adding 400 to the current bit address, we advance to the next segment's equivalent dot. We add 400 (not 800) since the odd and even dots are separated in the buffer. We do this firstly SegmentGroups sets of SegmentsPerXfer times to retrieve the data representing the even dots (the dot data is transferred to the MJI 36 bits at a time) and another SegmentGroups sets of SegmentsPerXfer times to load the odd dots. This entire process is repeated 400 times, incrementing the start address each time. Thus all dot values are transferred in the order required by the printhead in 400 × 2 × SegmentGroups × SegmentsPerXfer cycles.

In addition, we generate the TransferWriteEnable control signal. Since the LLFU starts before the MJI, we must transfer the first value before the Advance pulse from the MJI. We must also generate the next value in readiness for the first Advance pulse. The solution is to transfer the first value to the Transfer register after SegmentsPerXfer cycles, and then to stall SegmentsPerXfer-cycles later, waiting for the Advance pulse to start the next SegmentsPerXfer cycle group. Once the first Advance pulse arrives, the LLFU is synchronized to the MJI. However, the LineSync pulse to start the next line must arrive at the MJI at least 2SegmentsPerXfer cycles after the LLFU so that the initial Transfer value is valid and the next 32-bit value is ready to be loaded into the Transfer register.

The read process is shown in the following pseudocode:

```
DoneFirst = FALSE
For DotInSegment0 = 0 to 400
   CurrAdr = DotInSegment0
   XfersRemaining = 2 x SegmentGroups
   DotCount = SegmentsPerXfer
   Do
      V1 = DotCount = 0
      TransferWriteEnable = (V1 AND NOT DoneFirst) OR ADVANCE
      Stall = V1 AND (NOT TransferWriteEnable)
      If (NOT Stall)
         Shift Register=Fetch 4-bits from CurrReadBuffer:CurrAdr
         CurrAdr = CurrAdr + 400
         If (V1)
            DotCount = SegmentsPerXfer - 1
            XfersRemaining = XfersRemaining - 1
         Else
            DotCount = DotCount - 1
         EndIf
      EndIf
   Until (XfersRemaining=0) AND (NOT Stall)
EndFor
```

The final transfer may not be fully utilized. This occurs when the number of segments per transfer does not divide evenly into the actual number of segments in the printhead. An example of this is the 8½" printhead, which has 17 segments. Transferring 9 segments each time means that only 8 of the last 9 segments will be valid. Nonetheless, the timing requires the entire 9th segment value to be generated (even though it is not used). The actual address is therefore a don't care state since the data is not used.

Once the line has finished, the CurrReadBuffer value must be toggled by the processor.

15.4.2.2 Writing

The write process is also straightforward. 4 DMA request lines are output to the DMA controller. As requests are satisfied by the return DMA Acknowledge lines, the appropriate 8-bit destination address is selected (the lower 5 bits of the 15-bit output address are *don't care* values) and the acknowledge signal is passed to the correct buffer's WriteEnable control line (the Current Write Buffer is ¬CurrentReadBuffer). The 10-bit destination address is selected from the 4 current addresses, one address per color. As DMA requests are satisfied the appropriate destination address is incremented, and the corresponding TransfersRemaining counter is decremented. The DMA request line is only set when the number of transfers remaining for that color is non-zero.

The following pseudocode illustrates the Write process:

```
CurrentAdr[1-4] = 0
While (ColorXfersRemaining[1-4] are non-zero)
   DMARequest[1-4] = ColorXfersRemaining[1-4] NOT = 0
   If DMAAknowledge[N]
      CurrWriteBuffer:CurrentAdr[N] = Fetch 32-bits from data bus
      CurrentAdr[N] = CurrentAdr[N] + 1
      ColorXfersRemaining[N] = ColorXfersRemaining[N] - 1 (floor 0)
   EndIf
EndWhile
```

15.4.3 Registers

The following interface registers are contained in the LLFU:

Table 21. Line Load/Format Unit Registers

SegmentsPerXfer	The number of segments whose dots must be loaded before each transfer. This has a maximum value of 9.
SegmentGroups	The number of segment groups in the printhead. This has a maximum number of 4.
CurrentReadBuffer	The current buffer being read from. When Buffer0 is being read from, Buffer1 is written to and vice versa. Should be toggled with each AdvanceLine pulse from the MJI.
Go	Bits 0 and 1 control the starting of the read and write processes respectively. A non-zero write to the appropriate bit starts the process.
Stop	Bits 0 and 1 control the stopping of the read and write processes respectively. A non-zero write to the appropriate bit stops the process.
Stall	This read-only status bit comes from the LLFU's Stall flag. The Stall bit is valid when the write Go bit is set. A Stall value of 1 means that the LLFU is waiting for the ADVANCE pulse from the MJI to continue. The CPU can safely start the LSGU for the first line once the Stall bit is set.
ColorXfersRemaining[1-4]	The number of 32-bit transfers remaining to be read into the specific Color[N] buffer.

15.5 CONTROLLING A PRINT

When controlling a print the CPU programs and starts the LLFU in read mode to ensure that the first line of the page is transferred to the buffer. When the interrupts arrive from the DMA controller, the CPU can switch LLFU buffers, and program the MJI. The CPU then starts the LLFU in read/write mode and starts the MJI. The CPU should then wait a sufficient period of time to ensure that other connected printer controllers have also started their LLFUs and MJIs (if there are no other connected printer controllers, the CPU must wait until the Stall bit of the LLFU is set, a duration of 2SegmentsPerXfer cycles). The CPU can then program the LGSU to start the synchronized print. As interrupts arrive from the DMA controllers, the CPU can reprogram the DMA channels, swap LLFU buffers, and restart the LLFU in read/write mode. Once the LLFU has effectively filled its pipeline, it will stall until the next Advance pulse from the MJI. The MJI does not have to be touched during the print.

If for some reason the CPU wants to make any changes to the MJI or LLFU registers during an inter-line period it should ensure that the current line has finished printing/loading by polling the status bits of the MJI and the Go bits of the LLFU.

REFERENCES

16 Silverbrook References

[1] Silverbrook Research, *Authentication Chip*, 1998
[2] Silverbrook Research, *Authentication of Consumables*, 1998
[3] Silverbrook Research, *CePrint Product Design Description*, 1998
[4] Silverbrook Research, *Memjet*, 1998
[5] Silverbrook Research, *Netpage Pen Architecture*, 1999

17 Other References

[6] ANSI/EIA 538-1988, *Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Equipment*, August 1988

[7] Farrell, J., "How to Allocate Bits to Optimize Photographic Image Quality", Proceedings of IS&T International Conference on Digital Printing Technologies, 1998, pp.572-576

[8] Humphreys, G.W., and V. Bruce, *Visual Cognition*, Lawrence Erlbaum Associates, 1989, p.15

[9] ISO/IEC 19018-1:1994, *Information technology - Digital compression and coding of continuous-tone still images: Requirements and guidelines*, 1994

[10] Loeffler, C., A. Ligtenberg and G. Moschytz, "Practical Fast 1-D DCT Algorithms with 11 Multiplications", Proceedings of the International Conference on Acoustics, Speech, and Signal Processing 1989 (ICASSP '89), pp.988-991

[11] Thompson, H.S., *Multilingual Corpus 1* CD-ROM, European Corpus Initiative

[12] Urban, S.J., "Review of standards for electronic imaging for facsimile systems", Journal of Electronic Imaging, Vol.1(1), January 1992, pp.5-21

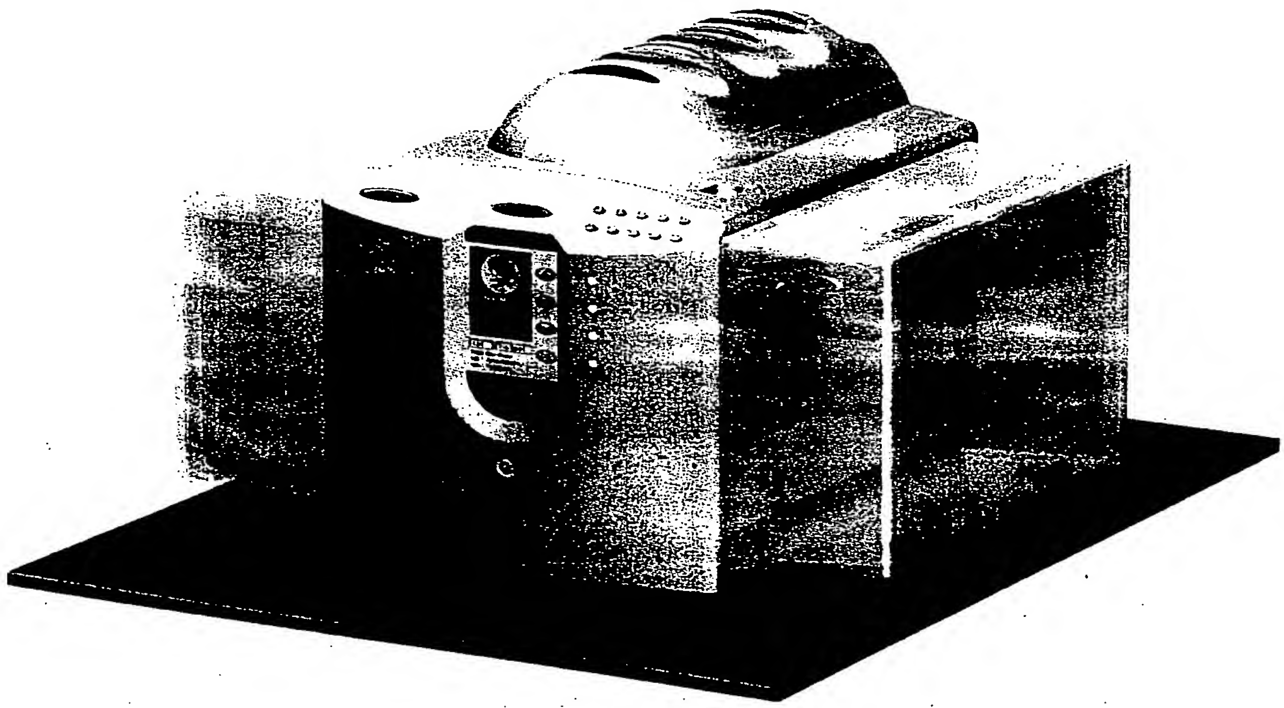
[13] Wallace, G.K., "The JPEG Still Picture Compression Standard", Communications of the ACM, Vol.34, No.4, April 1991, pp.30-44

Appendix 4
(Pages 22)

S-print Product Concept

DUPLEX NETWORK COLOR PRINTER
120 PPM
1600 DPI PHOTOGRAPHIC QUALITY

DRAFT VERSION 0.2, 22 MAY 1999



Silverbrook Research Pty Ltd
393 Darling Street, Balmain
NSW 2041 Australia
Phone: +61 2 9818 6633
Fax: +61 2 9818 6711
Email: info@silverbrook.com.au

CONFIDENTIAL

1 S-print Overview

S-print is a high-speed duplex network color printer intended for high-volume office use. It features 2000-sheet motorized paper trays, 120 page-per-minute operation, and 1600 dpi photographic-quality output.

With 20 times the speed of the best network color laser printers, and 4 times the speed of the best network monochrome laser printers, S-print effectively targets the $40 billion desktop laser printer market[1]. With its high performance and photographic-quality output, it also competes against offset printing for print runs smaller than 5000 copies.

S-print accommodates A4/Letter sized media and, with a tray adapter, A3/Tabloid sized media. It achieves simultaneous high quality and performance using full-color page-width 1600 dpi Memjet printheads.

S-print uses an embedded DSP-based raster image processor (RIP) to rasterize Postscript and PCL page descriptions at high speed. The standard RIP uses a single DSP, but up to three additional DSP modules can be plugged in to increase performance.

The RIP compresses and stores the rasterized page images on an internal high-capacity hard disk. While simple page descriptions are rasterized at the full 120 ppm printing rate, more complex page descriptions may take longer. Pre-rasterized documents retrieved from the internal hard disk are always printed at the full 120 ppm printing rate. Any document can be "printed" to the hard disk for later high-speed retrieval.

Users can walk up to an S-print, select locally-stored documents on its color LCD, and print them immediately, without ever going near a workstation. Documents printed in this way always print at the full 120 ppm rate. The standard 14GB internal hard disk stores over 6000 image-intensive pages. Because of its walk-up capability and high speed, S-print is likely to displace many uses of short-run offset printing.

S-print uses duplexed printheads for simultaneous double-sided printing. During the pilot phase of Memjet printhead manufacturing when the printhead defect density is still potentially high, each printhead is replicated to achieve 2:1 nozzle redundancy. This allows factory-detected defective nozzles to be bypassed, and so maximises printhead yield. A pair of custom print engines expand, dither and print page images to the duplexed printheads in real time.

Apart from custom print engines and Memjet printheads, S-print is built using standard off-the-shelf electronic components.

1. 1998 market size (source: IT Strategies)

2 Network Printer Comparison

Table 1 compares the specifications and running costs of S-print with those of various Hewlett-Packard network laser printers, ranging from the entry-level HP 4500 to the high-end HP 8500, and including the high-end monochrome HP 8100. Product descriptions for these HP printers appear at the end of this document (from www.hp.com).

Table 1. Network printer comparison

Specifications				
Available	Projected 2001	Now	Now	Now
Speed (color)	120 ppm	4 ppm	6 ppm	NA
Speed (monochrome)	120 ppm	16 ppm	24 ppm	32 ppm
Technology	Liquid Inkjet	Laser EP	Laser EP	Laser EP
Dot pitch	1600 dpi	600 dpi	600 dpi	600 dpi
Configuration	Desktop	Desktop	Floor-standing	Desktop
Duty cycle (pages per month)	600,000	35,000	60,000	150,000
Duplex	Yes	Option	Option	Option
Full bleed (prints to edge of paper)	Yes	No	No	No
Electronic collation	Yes	Option	Option	Option
Walk-up printing of stored documents	Yes	No	No	Option
Paper path	Straight	Complex	Complex	Complex
Color model	CMYK	CMYK	CMYK	Monochrome
Colorant	Water-based ink	Dry toner	Dry toner	Dry toner
Network-ready	Yes	Option	Option	Option
Standard paper capacity	2000	400	1100	1100
Optional maximum paper capacity	2000	900	3100	3100
Maximum paper size	A3/Tabloid	A4/Letter	A3/Tabloid	A3/Tabloid
Raster image processor (RIP)	1-4 Parallel DSPs	133 MHz MIPS 4700	133 MHz MIPS 4700	166 MHz NEC VR4310
Memory (standard)	64 MB	32 MB	32 MB	16 MB
Memory (maximum)	256 MB	208 MB	256 MB	208 MB
Disk drive	14 GB	Optional 2.1 GB	Optional 2.1 GB	Optional 2.1 GB
Control panel	Color TFT LCD	2-line alpha LCD	2-line alpha LCD	2-line alpha LCD
Printer languages	Postscript, PCL	Postscript, PCL	Postscript, PCL	Postscript, PCL
Fonts	TrueType, Postscript	TrueType, Postscript	TrueType, Postscript	TrueType, Postscript
Warm-up time	Instant	5 minutes	5 minutes	1.5 minutes
Power consumption (color)	300 Watts	300 Watts	375 Watts	NA
Power consumption (monochrome)	75 Watts	500 Watts	750 Watts	700 Watts
Power consumption (standby)	2 Watts	140 Watts	220 Watts	145 Watts
Power consumption (15% coverage)	40 Watts	300 Watts	300 Watts	700 Watts
Dimensions (H x W x D, imperial)	13.4" x 20.5" x 14.6"	15.4" x 19.7" x 22.4"	41.7" x 26.3" x 28.1"	22.3" x 20.5" x 21.6"
Dimensions (H x W x D, metric)	340 x 520 x 370 mm	390 x 500 x 570 mm	1060 x 668 x 714 mm	566 x 520 x 540 mm
Weight	20 kg (44 lb)	50.8 kg (112 lb)	101.2 kg (223 lb)	54 kg (120 lb)
Printer cost				
Minimum system	$3,000	$2,499	$5,999	$2,679
Optioned system[b]	$6,000	$4,674	$7,895	$5,774
Consumables capacity				
Black ink/toner (5% coverage)	88,778 pages	9,000 pages	17,000 pages	20,000 pages

Table 1. Network printer comparison

Color ink/toner (5% coverage)	44,389 pages	6,000 pages	8,500 pages	NA
Drum kit	NA	6,500 pages	12,500 pages	NA
Transfer kit	NA	25,000 pages	75,000 pages	NA
Fuser kit	NA	50,000 pages	100,000 pages	NA
Consumables cost				
Black ink/toner	$100.00	$77.94	$136.00	$273.00
Color ink/toner	$150.00	$97.51	$190.00	NA
Drum kit	NA	$75.04	$187.00	NA
Transfer kit	NA	$174.00	$445.00	NA
Fuser kit	NA	$203.00	$360.00	NA
Cost per page (not including paper)				
Light (5% coverage)	1.13 cents	8.00 cents	9.96 cents	1.37 cents
Typical (15% coverage)	3.38 cents	19.48 cents	24.97 cents	4.10 cents
Full page photos	11.26 cents	59.67 cents	77.51 cents	13.65 cents
Annual consumables revenues[c]				
Pages printed	936,000	31,200	46,800	249,600
Revenue	$31,629	$6,078	$11,685	$10,221

a. All HP specifications and prices from www.hp.com, 13 April 1999.
b. Duplex, network-ready, 2000-sheet capacity, 64MB, (S-print: 4 DSPs).
c. If the printer is used for 1 hour per day, 5 days per week, at 15% coverage, with 50% RIP overhead. High-end users will likely exceed his, while the average will probably be lower.

The actual market in which S-print would compete is wider than the network color printer market, since Memjet-based printers can also undercut the cost-per-page of monochrome laser printers.

In the table, the S-print ink is priced to slightly undercut monochrome laser printers, and to undercut color laser printers by a factor of 6 to 8. Color laser consumables are complex, including toner, complex mechanical toner stirring, imaging drums, transfer systems, fusers, and various other accessories. It is therefore doubtful that the price of these consumables could be reduced by a similar factor and still be profitable.

S-print ink cartridges are highly profitable. The black ink cartridge contains 1 liter of ink, costs $5 to manufacture, and sells for $100. The color ink cartridge contains 0.5 liter of ink, costs $3.50 to manufacture, and sells for $150.

The S-print printer is priced between $3000 and $6000 depending on options. This makes it competitive with high-end monochrome laser printers such as the HP 8100. Just as color inkjet printers have displaced monochrome inkjet printers at the low end of the market, color is also likely to displace monochrome in the midrange once color matches or betters the speed and cost-per-page of monochrome.

In the annual consumables section at the end of the table, S-print is shown to print almost four times as many pages as the HP 8100. This is enabled by the fact that it prints four times faster than the HP 8100, and is justified by the likelihood that it will display much short-run offset-printing in favor of in-house printing.

3 Utilization of Pilot Phase Yield

Printhead manufacturing volume is predicted to go through three major phases:

1) An initial lab phase, where only around 100 wafers per month are processed, the yields are low to non-existent, and the available number of printheads is very limited. Defect densities above 100 per cm^2 are assumed.

2) A pilot phase, where the number of wafers processed each month is around 5000. Defect densities are assumed to be between 10 per cm^2 and 100 per cm^2, necessitating the use of redundancy to achieve good printhead yields.

3) A full manufacturing phase, where 25,000 wafers are processed each month. Here defect densities are expected to fall from around 8 per cm^2 to 0.1 per cm^2 over the course of five years. During this phase, it is most cost-effective to *not* use printhead redundancy.

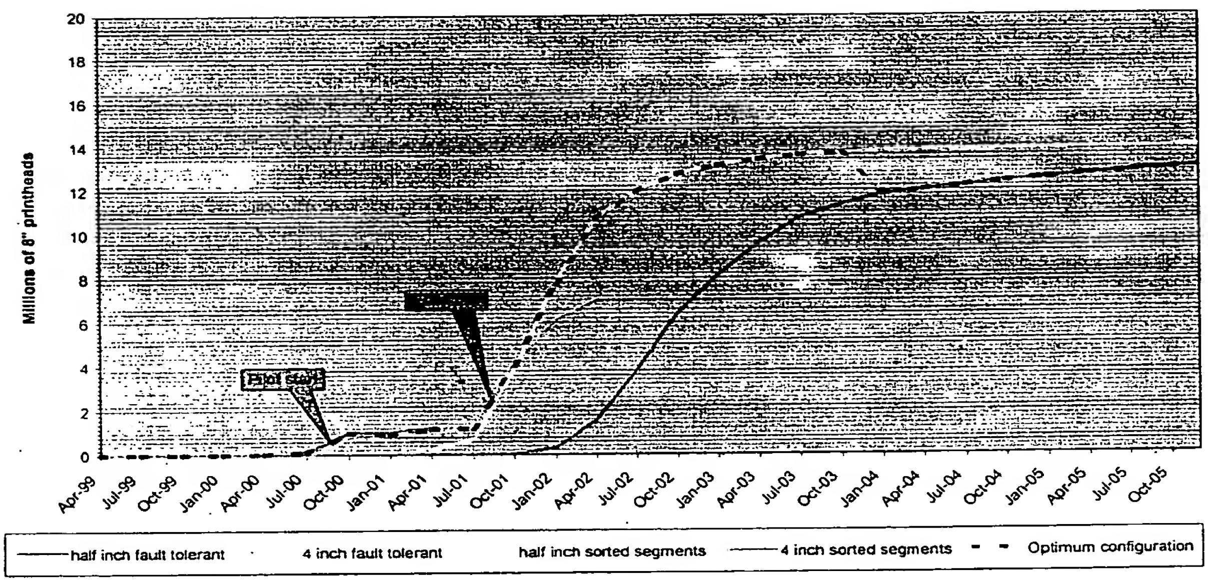


Figure 1. Millions of 8" printheads per quarter for various configurations (showing the optimum configuration)

Figure 1 shows manufacturing volume as a function of printhead configuration. The pilot phase, from October 2000 to October 2001, is characterized by a manufacturing volume of about one million deployable 8" printheads, made possible by redundant-nozzle printhead configurations. The full manufacturing phase, starting in October 2001, is characterized by a rapidly increasing manufacturing volume no longer reliant on nozzle redundancy.

The market focus during the pilot phase and the full manufacturing phase should be different. During the pilot phase, markets which are relatively insentitive to printhead cost should be targeted. These include:

1) Wide-format printing (machine prices around $10,000)

2) Network color printing (machine prices between $3000 and $7000)

3) Digital commercial printing (machine prices $100,000 and up)
4) Photo finishing (machine prices between $5000 and $20,000)

During the full manufacturing phase, high volume markets can also be targeted.

One million redundantly-deployable 8" printheads is sufficient for around 167,000 S-print printers.

Table 2 indicates the size of the various digital printing markets. Markets in which S-print is likely to compete effectively are highlit.

Table 2. Selected digital printing markets, 1998 (source: IT Strategies)

[illegible]	[illegible]	[illegible]	Media	Ink	[illegible]	Installed base
[illegible]						
Consumer & SOHO Ink Jet	$17,800 M	$9,000 M	$3,520 M	$5,280 M	29,000,000	81,000,000
Fax/MFP	$8,000 M	$4,000 M	$1,500 M	$2,500 M	5,000,000	12,500,000
Color Copier	$5,750 M	$2,000 M	$750 M	$3,000 M	45,000	183,000
Wide Format Graphic	$2,930 M	$500 M	$1,080 M	$1,350 M	44,000	84,000
Wide Format CAD	$1,733 M	$450 M	$513 M	$770 M	125,000	500,000
Bar Code/Direct Mail/Coding	$1,150 M	$750 M	$100 M	$300 M	14,000	75,000
[illegible]						
Kiosk Photo Printing	$241 M	$120 M	$48 M	$73 M	10,000	36,000
Card Printing (ID, Smart Cards)	$300 M	$101 M	$109 M	$90 M	18,000	51,000
Proofing Printing	$513 M	$250 M	$105 M	$158 M	11,000	72,000
MICR Printing	$205 M	$90 M	$50 M	$65 M	15,000	50,000
Medical Printing (X-ray, diagnostic)	$125 M	$35 M	$50 M	$40 M	3,000	15,000
Forms Printing	$110 M	$50 M	$20 M	$40 M	500	1,500
Label Printing	$13 M	$10 M	$2 M	$1 M	15,000	16,000
Total	$82,520 M	$34,251 M	$18,777 M	$29,492 M	42,820,500	134,083,500

a. S-print competes at the high end of this market, where the unit price of a printer is above $2000.

4 Printer Mechanics

S-print is compact, with minimum dimensions of 13.4" (340 mm) high by 20.5" (521 mm) wide by 14.6" (371 mm) deep (see Figure 2). As the print engine is small, the machine is dominated by paper and ink storage.

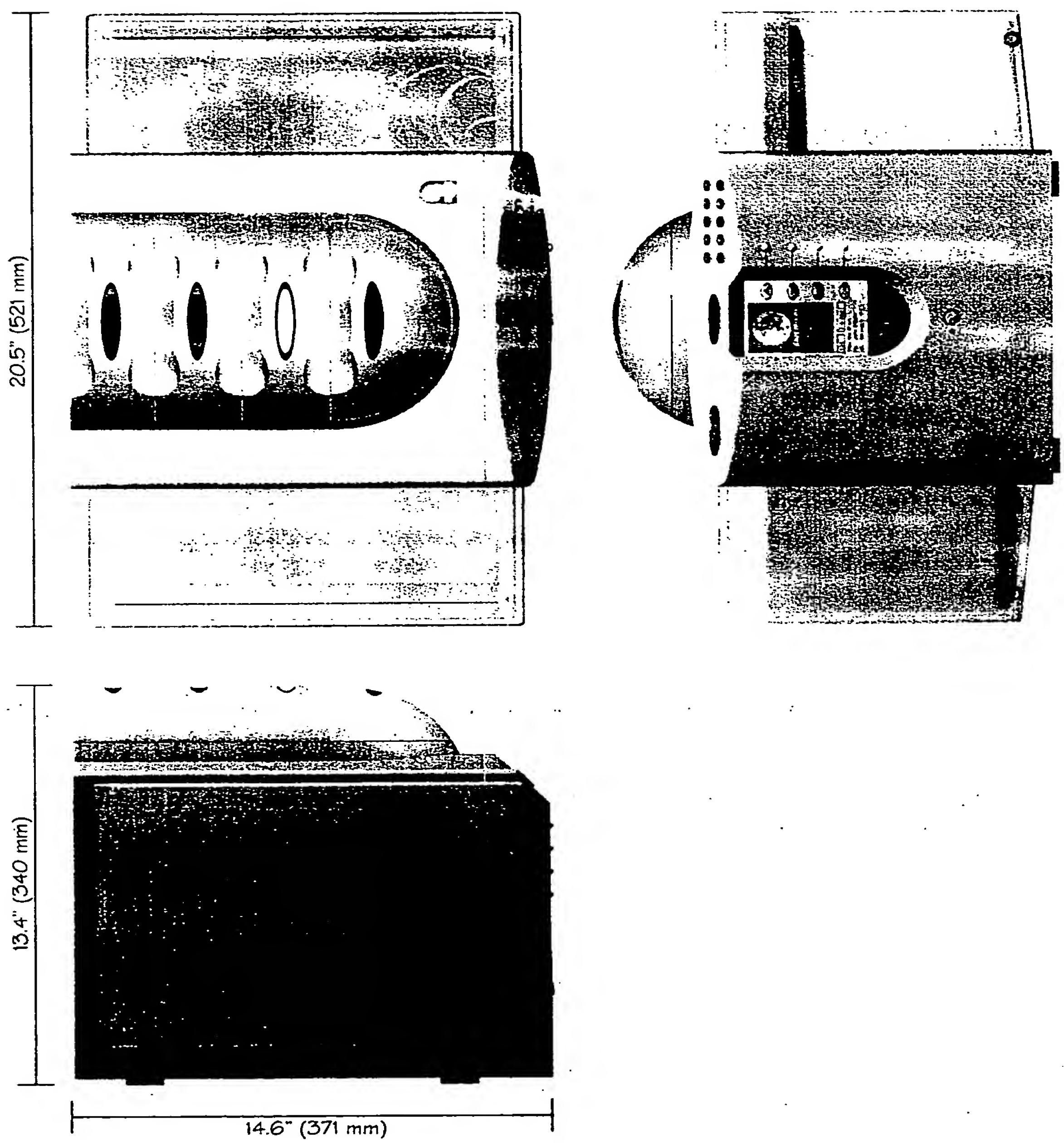


Figure 2. Plan and elevations of S-print printer unit

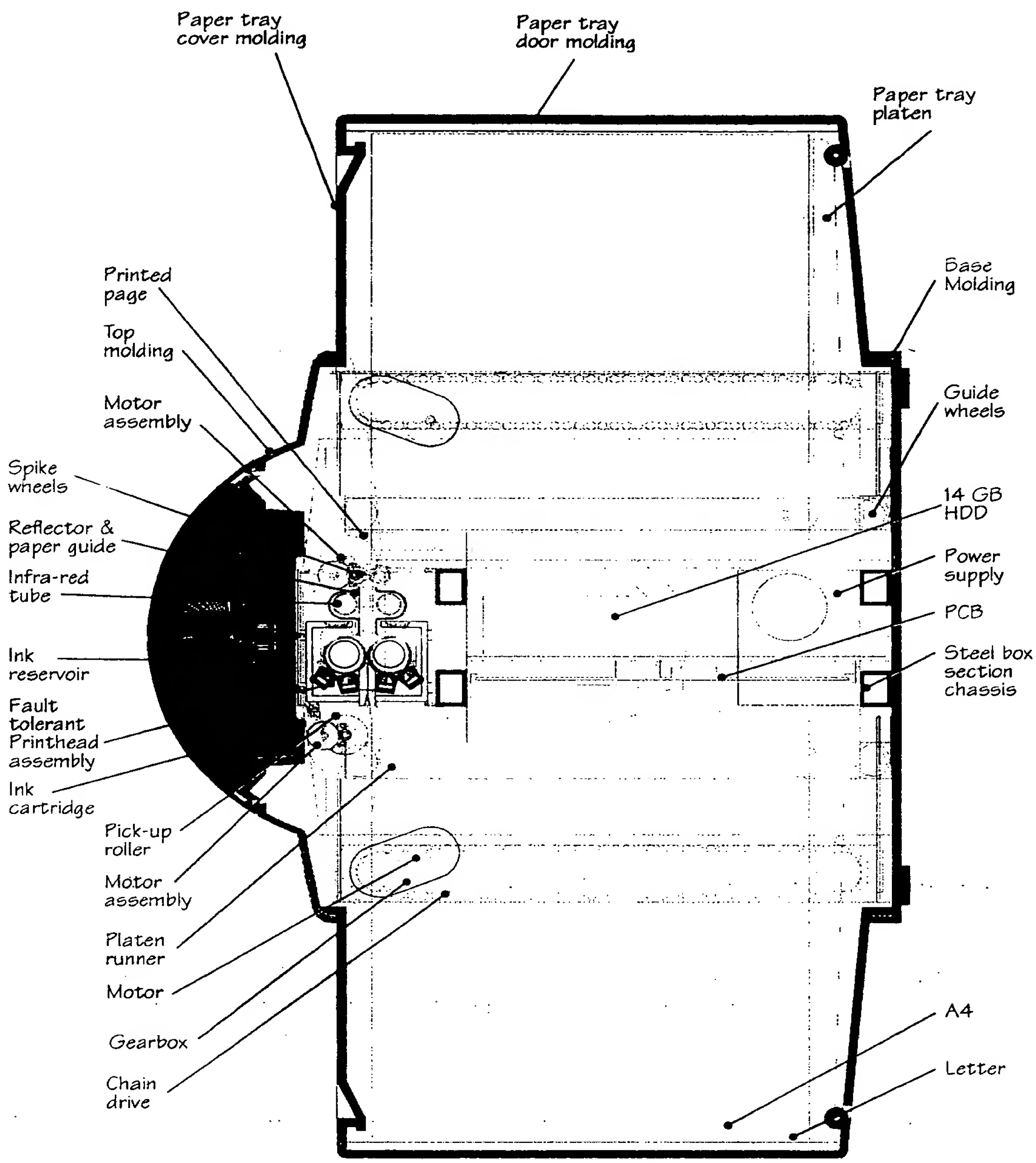


Figure 3. Front elevation of S-print (rotated)

The overall views of S-print are shown in Figure 3 and Figure 4. The elevation in Figure 3 shows the basic structure of the printer. It is built around a steel box section chassis with metal front, back and base panels attached. Two motorized paper trays form the basic paper path, with the imaging engines in the middle. The printer prints the wide edge of the paper to enable A3/Tabloid printing, and to achieve a compact form factor.

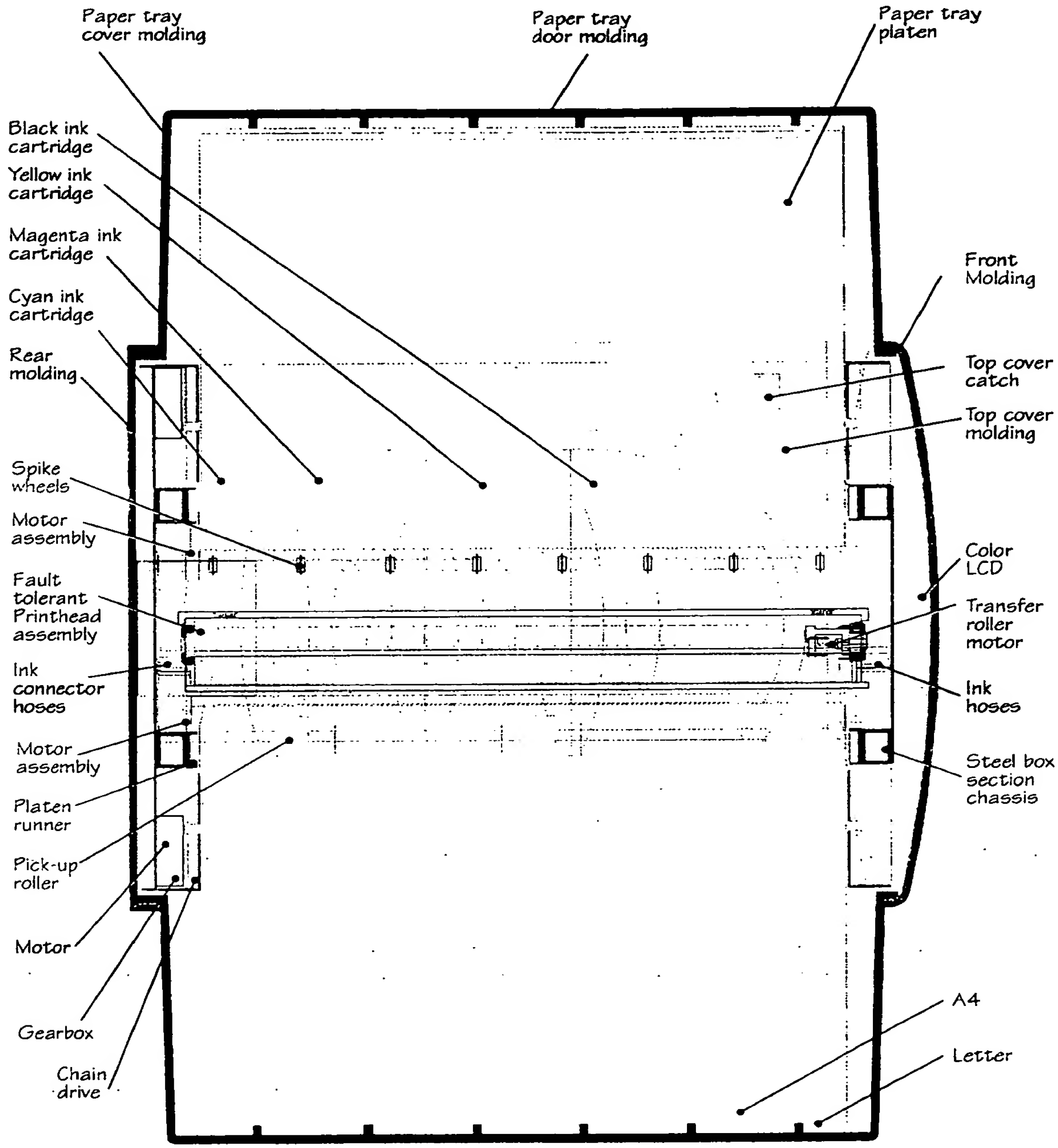


Figure 4. Plan of S-print

The trays comprise a metal platen with rollers, roller guides, a chain drive, motor and high ratio gearbox. The chain drive is duplicated on the opposite side of the platen and linked via an axle at base level. This is a proven mechanism that allows smooth lifting and lowering of up to 2000 sheets of paper. A metal plate on one side of the paper tray can be relocated to provide either A4 or Letter width guidance. This entire paper tray assembly is duplicated on the other side of the printer to accept the high speed paper flow. Each tray is

housed in an injection-molded, color-tinted, transparent housing with a hinged door and latch sensors.

Another motor assembly powers the paper pick-up roller which feeds the sheet through to the imaging engines. The top molding in Figure 5 can be unlatched and hinged upwards to reveal the metalwork chassis for the upper imaging engine. This in turn can be unlocked and pivoted upward for access to the printheads and drying lamps.

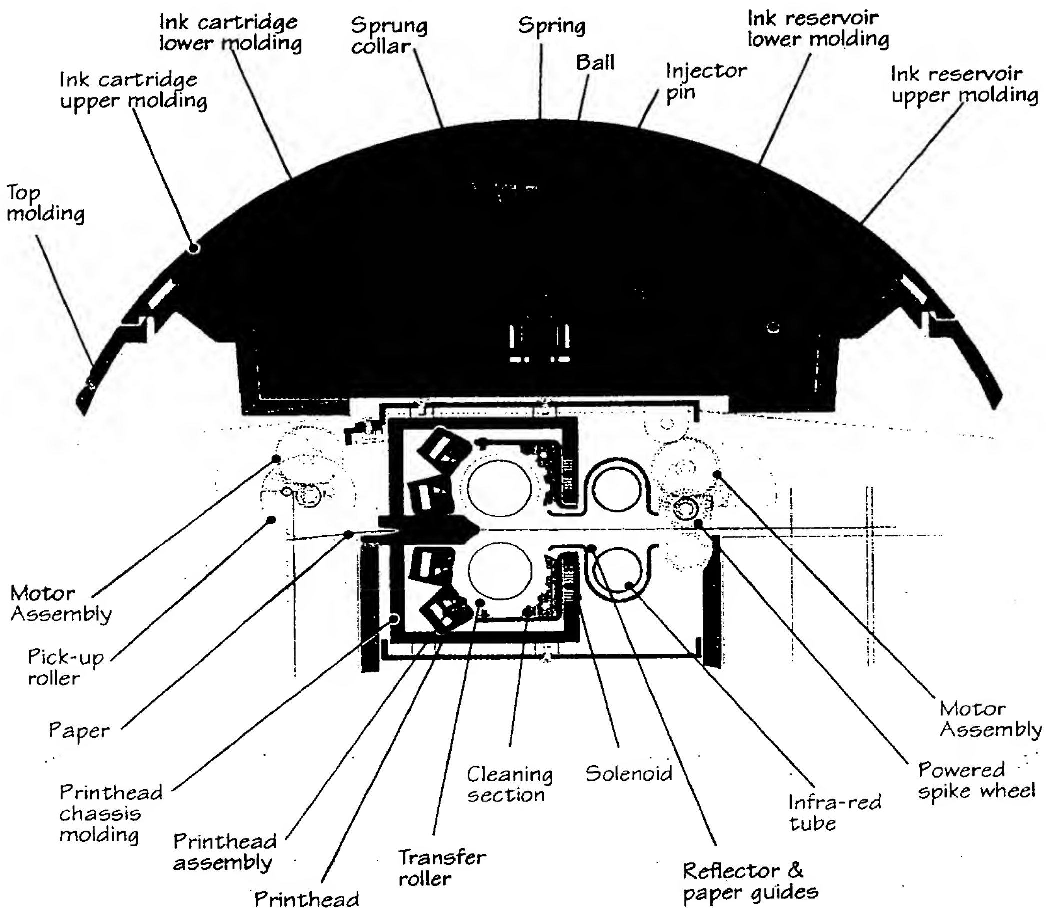


Figure 5. Ink supply and paper path between duplexed imaging engines

Figure 5 shows the workings of S-print. The straight paper path allows the paper to be fed at high speed past the printheads. The two imaging engines are mounted together in an adjustable assembly. The top imaging engine can pivot upwards to allow access to paper jams and to the lower imaging engine and infra-red drying lamps.

The transfer roller caps the printheads whenever S-print is not printing. Whenever a page is to be printed, the solenoid is activated, and the roller pivots away from the printheads and contacts against the transfer roller from the other imaging engine providing traction onto the paper.

The paper then passes the infra-red tubes which dry the ink before a set of spike wheels eject the printed page into the exit tray.

The ink cartridges snap fit onto the top molding. Ink is automatically fed via a sprung collared pin into a permanent ink reservoir. The reservoir serves to provide an early warning to replace the ink cartridge and makes the contact with the embedded QA cartridge chip. The provision of a large reservoir means that the ink cartridges typically only need to be replaced when all four colors are empty, rather than when the first color is empty. Each ink color reservoir connects via hoses to the printhead extrusion assemblies, which are daisy chained together via flexible hose connections (Figure 4).

The main PCB, power supply and 14GB hard disk are shown in Figure 3. Access to these components is via the front and rear moldings.

Figure 6. Front panel user interface

Figure 6 shows the color LCD interface with four changeable function buttons for navigating the user interface. The LCD displays a preview of the front page of the document to be printed. The keypad on the front molding allows the desired number of copies to be entered. Documents to be printed locally can also be selected by an ID number using the keypad. This can be quicker than scrolling through stored documents if there are many.

4.1 IMAGING ENGINES

S-print uses duplexed imaging engines (Figure 7) for simultaneous double-sided printing. During the pilot phase of manufacturing when the printhead defect density is still potentially high, each printhead is replicated to achieve 2:1 nozzle redundancy. This allows factory-detected defective nozzles to be bypassed, and so maximises printhead yield.

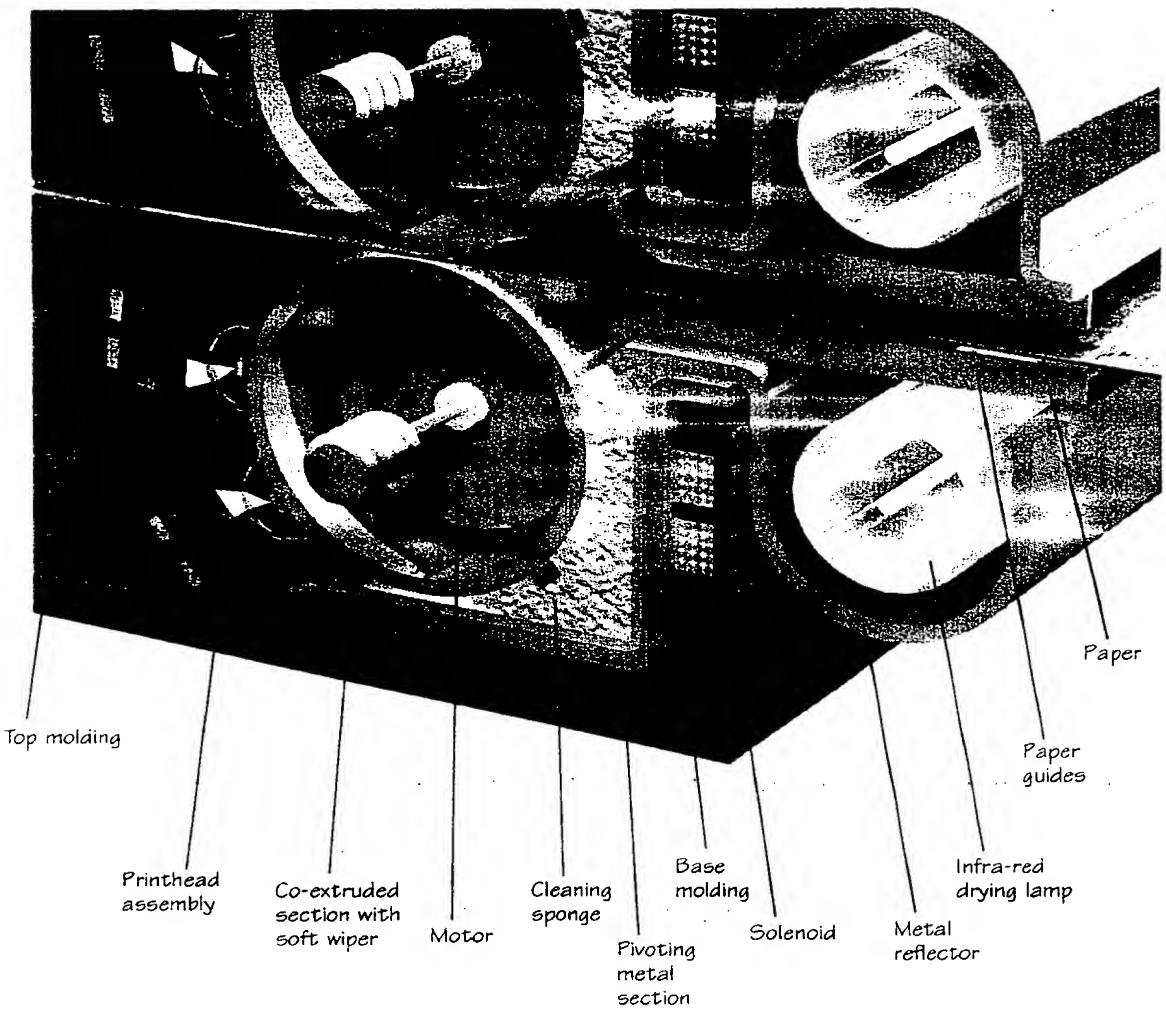


Figure 7. Duplexed imaging engines and drying lamps

Each imaging engine therefore contains two printhead assemblies (Figure 8) mounted side-by-side, sharing the same powered transfer roller and cleaning station.

The four printhead assemblies in the two imaging engines share the same ink supply.

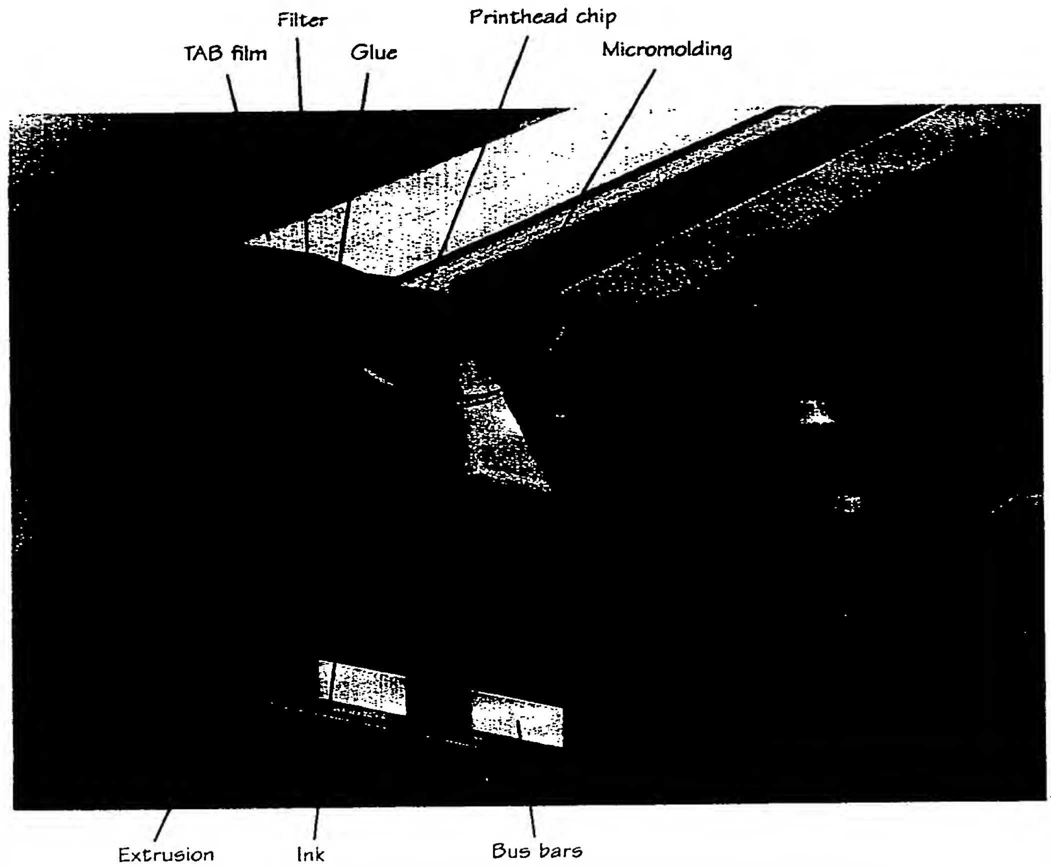


Figure 8. Cross-section of printhead assembly

5 Ink Cartridge

The black cartridge holds 1 liter (33.8 fluid ounces[1]), while the cyan, magenta and yellow cartridges each hold half a liter. The cartridges plug directly into the top of the printer and are comprised of two main moldings ultrasonically welded together with a QA chip in the lower molding (Figure 9).

The upper molding houses a sprung ball-bearing that is held captive against the inner molding. This provides the main seal to the cartridge with a secondary hydrophobic elastomeric seal to the entry port on the lower molding.

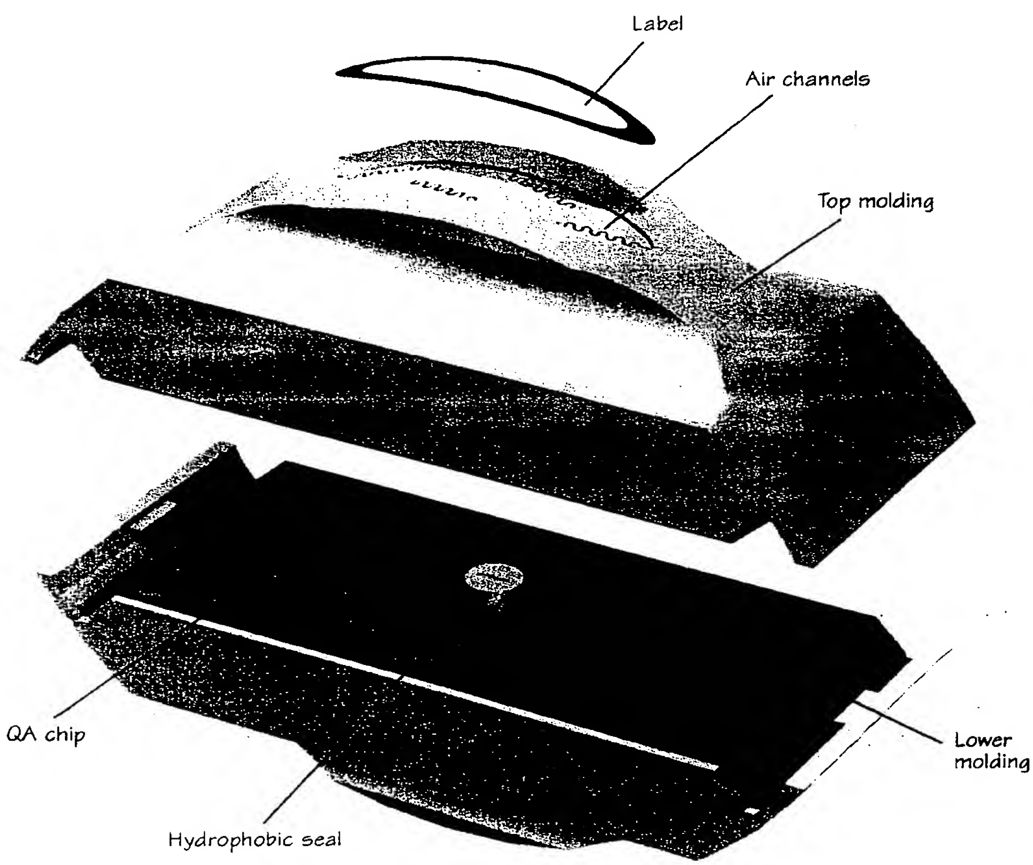


Figure 9. Ink cartridge explosion

The cartridge connects to the printheads via the printer ink reservoir. This has a hollow pin that emerges from a sprung sleeve when the cartridge is pushed on. The pin pushes the ball bearing back and allows ink to flow. Tortuous hydrophobic air channels are provided at the top of the unit under the label. The four ink cartridges are differently keyed by plastic protrusions to prevent any incorrect insertion or orientation of the units.

1. 2.11 US pints, 1.76 British pints, or 0.0042 hogsheads.

6 Memjet-Based Printing

A Memjet printhead produces 1600 dpi bi-level CMYK. On low-diffusion paper, each ejected drop forms an almost perfectly circular 22.5μm diameter dot. Dots are easily produced in isolation, allowing dispersed-dot dithering to be exploited to its fullest. Since the Memjet printhead is the width of the page and operates with a constant paper velocity, the four color planes are printed in perfect registration, allowing ideal dot-on-dot printing. Since there is consequently no spatial interaction between color planes, the same dither matrix is used for each color plane. Dot-on-dot printing minimizes 'muddying' of mid-tones caused by inter-color bleed.

A page layout may contain a mixture of images, graphics and text. Continuous-tone (contone) images and graphics are reproduced using a stochastic dispersed-dot dither. Unlike a clustered-dot (or amplitude-modulated) dither, a *dispersed-dot* (or frequency-modulated) dither reproduces high spatial frequencies (i.e. image detail) almost to the limits of the dot resolution, while simultaneously reproducing lower spatial frequencies to their full color depth, when spatially integrated by the eye. A *stochastic* dither matrix is carefully designed to be free of objectionable low-frequency patterns when tiled across the image. As such its size typically exceeds the minimum size required to support a particular number of intensity levels (e.g. 16×16×8 bits for 257 intensity levels). S-print uses a dither *volume* of size 64×64×3×8 bits. The volume provides an extra degree of freedom during the design of the dither by allowing a dot to change states multiple times through the intensity range (rather than just once as in a conventional dither matrix).

Human contrast sensitivity peaks at a spatial frequency of about 3 cycles per degree of visual field and then falls off logarithmically, decreasing by a factor of 100 beyond about 40 cycles per degree and becoming immeasurable beyond 60 cycles per degree. At a normal viewing distance of 12 inches (about 300mm), this translates roughly to 200-300 cycles per inch (cpi) on the printed page, or 400-600 samples per inch according to Nyquist's theorem.

In practice, contone resolution above about 300 ppi is of limited utility outside special applications such as medical imaging. Offset printing of magazines, for example, uses contone resolutions in the range 150 to 300 ppi. Higher resolutions contribute slightly to color error through the dither.

Black text and graphics are reproduced directly using bi-level black dots, and are therefore not antialiased (i.e. low-pass filtered) before being printed. Text is therefore *supersampled* beyond the perceptual limits discussed above, to produce smoother edges when spatially integrated by the eye. Text resolution up to about 1200 dpi continues to contribute to perceived text sharpness (assuming low-diffusion paper, of course).

S-print uses a contone resolution of 320 ppi (i.e. 1600÷5), and a black text and graphics resolution of 1600 dpi.

7 Document Data Flow

Document transmission and document rasterization are decoupled to shield the user from interactions between the size and complexity of the document, and the memory capacity and RIP performance of S-print. This is achieved by storing each document's PDL file on the internal hard disk.

Because of the high resolution of the Memjet printhead, each page must be printed at a constant speed to avoid creating visible artifacts. This means that the printing speed can't be varied to match the input data rate. Document rasterization and document printing are therefore decoupled to ensure the printhead has a constant supply of data. A page is never printed until it is fully rasterized. This is achieved by storing a compressed version of each rasterized page image on the internal hard disk.

This decoupling also allows the RIP to run ahead of the printer when rasterizing simple pages, buying time to rasterize more complex pages.

The user indicates whether a document is to be stored permanently on the hard disk, printed, or both. So long as there is disk space available, the pages of transient documents are also cached on the disk until printed. This is particularly efficient when multiple copies of complex documents are being printed. This so-called electronic collation also obviates the need for an external collating mechanism, since each copy of a document is printed in its entirety before the next copy.

Because contone color images are reproduced by stochastic dithering, but black text and line graphics are reproduced directly using black dots, the compressed page image format contains a separate foreground bi-level black layer and background contone color layer. The black layer is composited over the contone layer after the contone layer is dithered.

Figure 10 shows the flow of a S-print document from network to printed page.

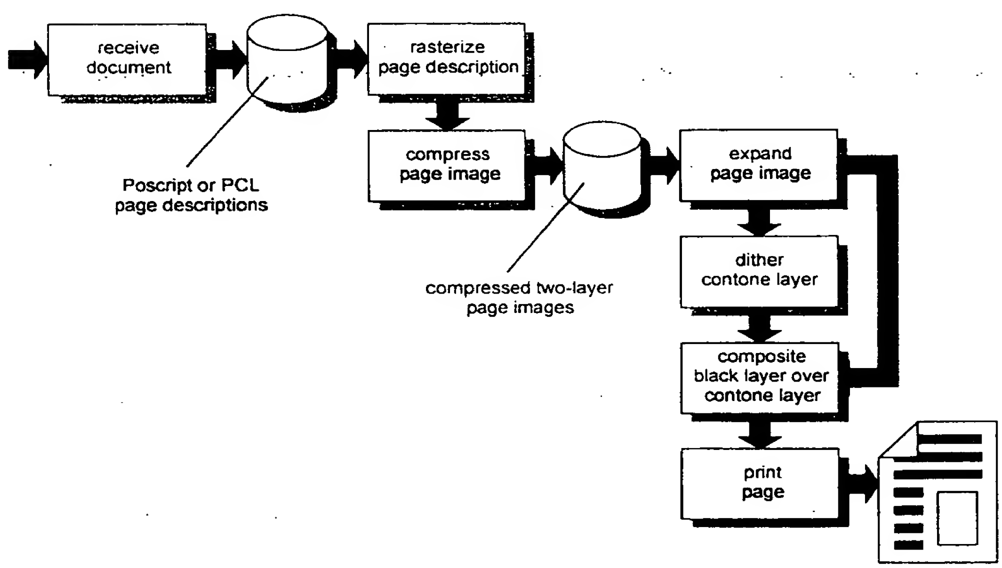


Figure 10. S-print document data flow

At 320 ppi, an A4/Letter page[1] of contone CMYK data has a size of 38MB. Using lossy contone compression algorithms such as JPEG, contone images compress with a ratio up to 10:1 without noticeable loss of quality, giving a compressed page size of 3.8MB.

At 1600 dpi, an A4/Letter page of bi-level data has a size of 30MB. Coherent data such as text compresses very well. Using lossless bi-level compression algorithms such as Group 4 Facsimile, ten-point text compresses with a ratio of about 20:1, giving a compressed page size of 1.5MB.

Once dithered, a page of CMYK contone image data consists of 120MB of bi-level data. Using lossless bi-level compression algorithms on this data is pointless precisely because the optimal dither is stochastic - i.e. since it introduces hard-to-compress disorder.

The two-layer compressed page image format therefore exploits the relative strengths of lossy JPEG contone image compression and lossless bi-level text compression. The format is compact enough to be storage-efficient, and simple enough to allow straightforward realtime expansion during printing.

Since text and images normally don't overlap, the normal worst-case page image size is 3.8MB (i.e. image-only), while the normal best-case page image size is 1.5MB (i.e. text-only). The absolute worst-case page image size is 5.3MB (i.e. text over image). Assuming a third of an average page contains images, the average page image size is 2.3MB. The standard 14GB internal hard disk therefore holds over 6000 such pages.

1. An A4/Letter page has a maximum area of 8.3"×11.7" (i.e. A4).

8 Printer Controller Architecture

The S-print printer controller consists of a controlling processor, various peripheral controllers, a raster image processor (RIP) DSP farm, and duplexed print engines. These components are discrete and communicate via a shared bus and a shared 64MB memory, as illustrated in Figure 11.

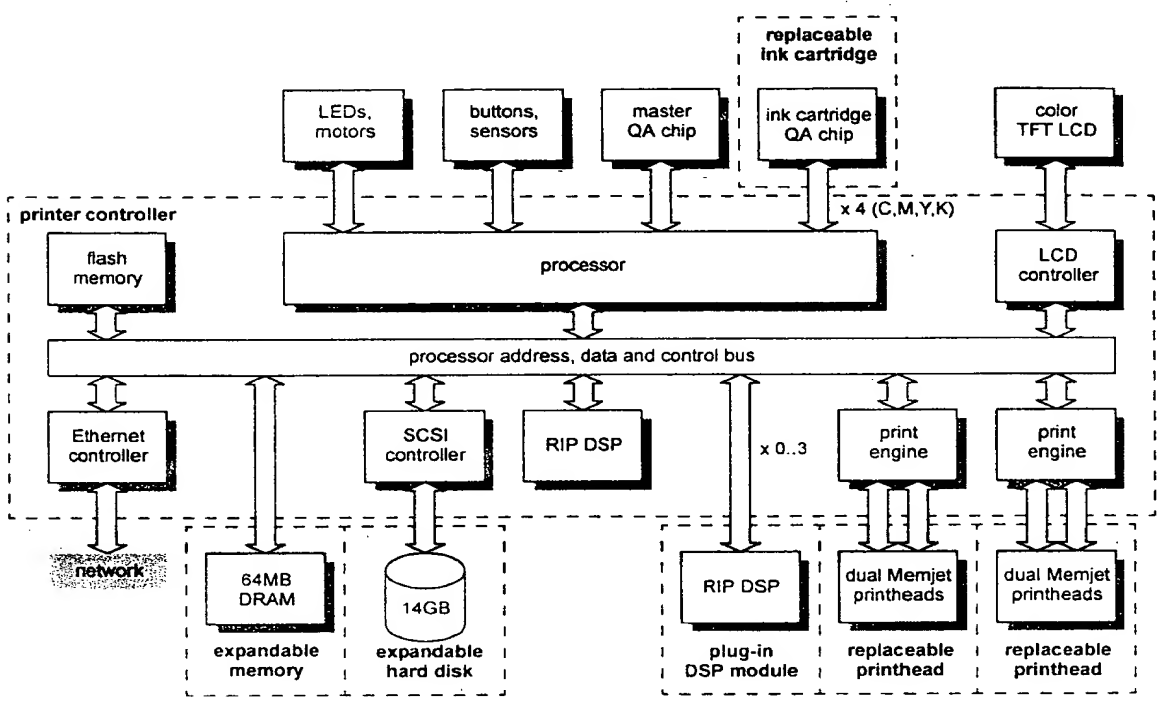


Figure 11. Printer controller architecture

The controlling processor handles communication with the network, controls the internal hard disk, controls the user interface (LCD, buttons and LEDs), controls the paper transport, handles ink cartridge authentication and ink monitoring, and feeds and synchronizes the RIP and the print engines. It consists of a medium-performance general-purpose microprocessor. Its associated peripheral controllers include a 10/100Base-T Ethernet controller, a SCSI disk controller, and a color TFT LCD controller. Optional controllers include an IEEE 1394 (Firewire) controller and a USB 2.0 controller for high-speed point-to-point communication with a workstation or server.

The RIP DSP farm rasterizes and compresses page descriptions to S-print's compressed page format. The DSP farm consists of between one and four general-purpose high-performance DSPs. Each additional DSP comes as a field-installable plug-in module.

Each print engine expands, dithers and prints page images to its associated replicated printhead in real time (i.e. at 60 ppm). The duplexed print engines print both sides of the page simultaneously (i.e. at 120 ppm).

The printer controller's flash memory holds the software for both the processor and the DSPs. This is copied to main memory at boot time. The flash memory also holds the defect characterization vectors for the two replicated printheads. These are copied to the print engines at boot time.

8.1 Detailed Document Data Flow

The main processor receives the document's PDL file and stores it on the internal hard disk. It then runs the appropriate RIP software on the DSPs.

The DSPs rasterize each page description and compress the rasterized page image. The main processor stores each compressed page image on the hard disk. The simplest way to load-balance multiple DSPs is to let each DSP rasterize a separate page. The DSPs can always be kept busy since an arbitrary number of rasterized pages can, in general, be stored on the internal hard disk. This strategy can lead to poor DSP utilization, however, when rasterizing short documents.

The main processor passes back-to-back page images to the duplexed print engines. Each print engine stores the compressed page image into its local memory, and starts the page expansion and printing pipeline. Page expansion and printing is pipelined because it is impractical to store a 120MB bi-level CMYK image in memory.

The first stage of the pipeline expands the JPEG-compressed contone CMYK layer. The second stage, in parallel with the first, expands the Group 4 Fax-compressed bi-level black layer. The third stage dithers the contone CMYK layer, and composites the bi-level black layer over the resulting bi-level CMYK layer. The fourth stage prints the bi-level CMYK data via the printhead interface which controls the Memjet printhead.

The main processor streams compressed page images from the hard disk to the print engines at the required 120 ppm rate (i.e. 4.6MB/s on average, or 10.6MB/s worst-case).

Table 3. Print engine page image and FIFO data flow

process	input format	input window	output format	output window	input rate	output rate
receive contone	-	-	JPEG stream	1	- -	3.8MB/s (10Mp/s)
receive bi-level	-	-	G4Fax stream	1	- -	1.5MB/s (250Mp/s)
expand contone	JPEG stream	-	32-bit CMYK	8	3.8MB/s (10Mp/s)	38MB/s (10Mp/s)
expand bi-level	G4Fax stream	-	1-bit K	1	1.5MB/s (250Mp/s)	30MB/s (250Mp/s)
dither	32-bit CMYK	1	-[a]	-	38MB/s (10Mp/s[b])	-
composite	1-bit K	1	4-bit CMYK	1	30MB/s (250Mp/s)	120MB/s (250Mp/s)
print	4-bit CMYK	24, 1[c]	-	-	120MB/s (250Mp/s)	- -
					193MB/s	193MB/s
					387MB/s	

a. dither combines with composite, so there is no external data flow between them
b. 320 ppi ⇒ 1600 dpi (5 × 5 expansion).

c. Needs a window of 24 lines, but only advances 1 line.

The print engine data flow is summarized in Table 3. The aggregate traffic to/from memory is 387MB/s, all but 5.3MB/s of which relates to the FIFOs.

Each stage communicates with the next via a FIFO. Each FIFO is organized into lines, and the minimum size (in lines) of each FIFO is designed to accommodate the output window (in lines) of the producer and the input window (in lines) of the consumer. The inter-stage memory FIFOs are described in Table 4.

Table 4. Print engine local memory FIFOs

FIFO	[illegible]	[illegible]	[illegible]
contone CMYK	32-bit interleaved CMYK (320 ppi x 11.7" x 32 = 15.0KB)	8 x 2 = 16	240KB
bi-level K	1-bit K (1600 dpi x 11.7" x 1 = 2.3B)	1 x 2 = 2	5KB
bi-level CMYK	4-bit planar odd/even CMYK (1600 dpi x 11.7" x 4 = 9.1KB)	24 + 1 = 25	229KB
			474KB

8.2 PRINT ENGINE ARCHITECTURE

The print engine is implemented as a single custom chip. It shares a common design with print engines used in other Memjet printers.

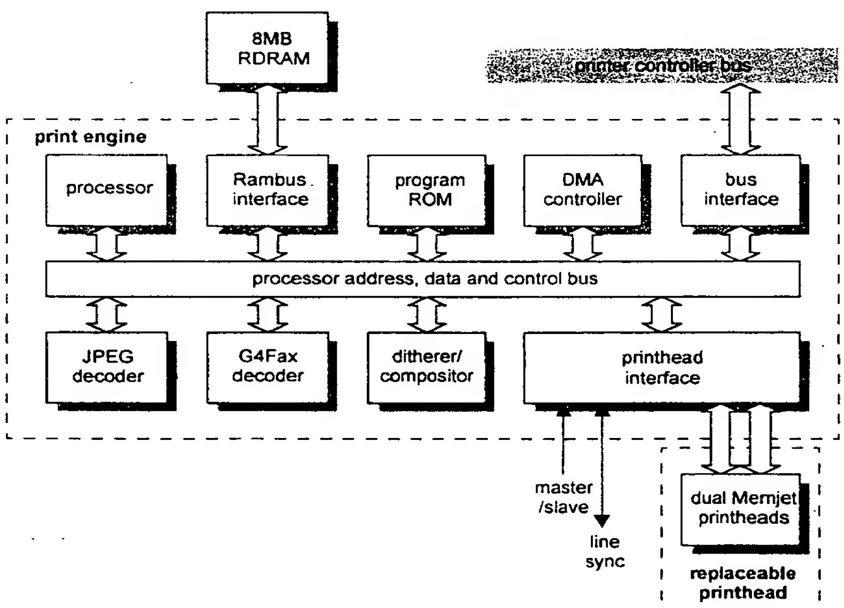


Figure 12. Shared-memory print engine architecture

There are two architectural variants of the print engine. The shared-memory version uses a local off-chip RDRAM to support the aggregate memory bandwidth required by page expansion and printing (see Figure 12). The pipelined version uses dedicated on-chip FIFOs (see Figure 13).

The shared-memory print engine consists of a general-purpose processor, a high-speed Rambus interface to the off-chip RDRAM, a small program ROM, a DMA controller, and an interface to the printer controller bus.

Both print engine's page expansion and printing pipeline consists of a standard JPEG decoder, a standard Group 4 Fax decoder, a custom ditherer/compositor unit, and a custom interface to the Memjet printheads. These are described in detail in design documents.

In the shared-memory version, the FIFOs are located in the dedicated off-chip RDRAM, and all inter-stage communication is controlled by the local processor via DMA. In the pipelined version, the FIFOs are on-chip, and the stages are self-synchronizing.

In the shared-memory version, the decoders obtain page data from the main processor via the local memory. In the pipelined version, the decoders obtain page data directly from the main processor over the printer controller bus.

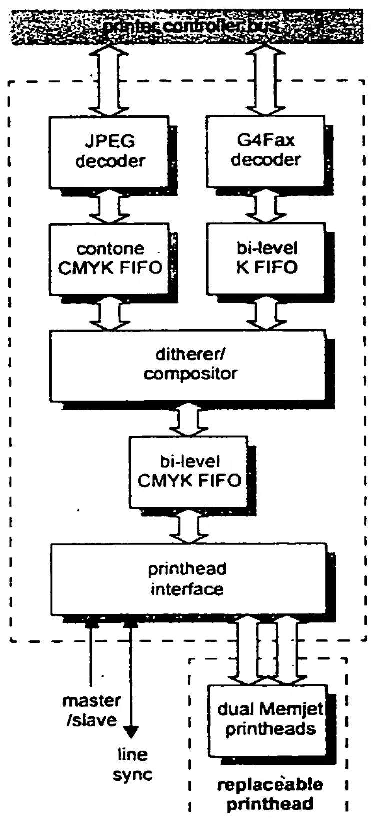


Figure 13. Pipelined print engine architecture

When several print engines are used in unison, such as in a duplexed configuration, they are synchronized via a shared line sync signal. Only one print engine, selected via the external master/slave pin, generates the line sync signal onto the shared line.

8.3 PRINTHEAD TIMING

Each print engine prints an A4/Letter page in one second. Since S-print uses a 12" printhead to print the long dimension of the page (11.7"), the printhead needs to traverse the short dimension of the page (8.5") in one second. At 1600 dpi, this equates to a 13.6KHz line rate. This is well within the operating frequency of the Memjet printhead, which in the current design exceeds 30KHz.

8.4 PRINTHEAD CHARACTERIZATION

Each redundant 12" printhead contains two complete 12" printheads, i.e. 76,800 nozzle pairs, characterized and matched so that no paired nozzles are both defective.

Printhead defects are either characterized and matched one segment at a time, or after the entire printhead has been built. In the former case nozzles are tested before integration with the ink path, and so are tested without ink. In the latter case nozzles are tested after integration with the ink path, and so are tested with ink. Segment-wise characterization gives a higher yield, but at a higher testing cost. It is therefore only preferable to printhead-wise characterization when defect densities are still high.

A characterization vector indicates, via one bit per nozzle pair, which nozzle of the pair is to be used. A 12" printhead characterization vector requires 9600 bytes to represent.

Once determined, the characterization vector associated with a redundant printhead is stored in the manufacturing database, indexed by the printhead's serial number, recorded as a barcode on its cartridge. When the printhead cartridge is finally inserted into a printer during manufacture, the characterization vector is retrieved using the barcode, and is written to the flash memory of the printer's embedded printer controller.

If the printhead cartridge is replaced in the field, then a new characterization vector is downloaded remotely from the manufacturing database to the printer controller via its network interface, using the new printhead cartridge's barcode.

New United States Patent Application
Inventors/Assignors: KIA SILVERBROOK and PAUL LAPSTUN
Assignee: SILVERBROOK RESEARCH PTY. LTD.
"METHOD AND SYSTEM FOR INSTRUCTION OF A
COMPUTER
Docket No. NPA047US